Fluent

高级应用与实例分析

江帆 黄鹏 编著



清华大学出版社

VISIT…

LANZAROTE Caliente.com

第 1 章　CFD 基础

计算流体动力学(computational fluid dynamics，CFD)是流体力学的一个分支，它通过计算机模拟获得某种流体在特定条件下的有关信息，实现了用计算机代替试验装置完成“计算试验”，为工程技术人员提供了实际工况模拟仿真的操作平台，已广泛应用于航空航天、热能动力、土木水利、汽车工程、铁道、船舶工业、化学工程、流体机械、环境工程等领域。

本章介绍 CFD 一些重要的基础知识，帮助读者熟悉 CFD 的基本理论和基本概念，为计算时设置边界条件、对计算结果进行分析与整理提供参考。

1.1　流体力学的基本概念

1.1.1　流体的连续介质模型

流体质点(fluid particle)：几何尺寸同流动空间相比是极小量，又含有大量分子的微元体。

连续介质(continuum/continuous medium)：质点连续地充满所占空间的流体或固体。

连续介质模型(continuum/continuous medium model)：把流体视为没有间隙地充满它所占据的整个空间的一种连续介质，且其所有的物理量都是空间坐标和时间的连续函数的一种假设模型：$u = u(t,x,y,z)$。

1.1.2　流体的性质

1. 惯性

惯性(fluid inertia)指流体不受外力作用时，保持其原有运动状态的属性。惯性与质量有关，质量越大，惯性就越大。单位体积流体的质量称为密度(density)，以 r 表示，单位为 kg/m^3。对于均质流体，设其体积为 V，质量为 m，则其密度为

$$\rho = \frac{m}{V} \tag{1-1}$$

对于非均质流体，密度随点而异。若取包含某点在内的体积 ΔV，其中质量 Δm，则该点密度需要用极限方式表示，即

$$\rho = \lim_{\Delta V \to 0} \frac{\Delta m}{\Delta V} \tag{1-2}$$

2. 压缩性

作用在流体上的压力变化可引起流体的体积变化或密度变化，这一现象称为流体的可压缩性。压缩性(compressibility)可用体积压缩率 k 来量度

$$k=-\frac{\mathrm{d}V/V}{\mathrm{d}p}=\frac{\mathrm{d}\rho/\rho}{\mathrm{d}p} \tag{1-3}$$

式中：p 为外部压强。

在研究流体流动过程中，若考虑到流体的压缩性，则称为可压缩流动，相应地称流体为可压缩流体，例如高速流动的气体。若不考虑流体的压缩性，则称为不可压缩流动，相应地称流体为不可压缩流体，如水、油等。

3. 粘性

粘性(viscosity)指在运动的状态下，流体所产生的抵抗剪切变形的性质。粘性大小由粘度来量度。流体的粘度是由流动流体的内聚力和分子的动量交换所引起的。粘度有动力粘度 μ 和运动粘度 ν 之分。动力粘度由牛顿内摩擦定律导出：

$$\tau=\mu\frac{\mathrm{d}u}{\mathrm{d}y} \tag{1-4}$$

式中：τ 为切应力，Pa；μ 为动力粘度，Pa · s；$\mathrm{d}u/\mathrm{d}y$ 为流体的剪切变形速率。

运动粘度与动力粘度的关系为

$$\nu=\frac{\mu}{\rho} \tag{1-5}$$

式中：ν 为运动粘度，$\mathrm{m^2/s}$。

在研究流体流动过程中，考虑流体的粘性时，称为粘性流动，相应的流体称为粘性流体；当不考虑流体的粘性时，称为理想流体的流动，相应的流体称为理想流体。

根据流体是否满足牛顿内摩擦定律，将流体分为牛顿流体和非牛顿流体。牛顿流体严格满足牛顿内摩擦定律且 μ 保持为常数。非牛顿流体的切应力与速度梯度不成正比，一般又分为塑性流体、假塑性流体、胀塑性流体 3 种。

塑性流体，如牙膏等，它们有一个保持不产生剪切变形的初始应力 τ_0，只有克服了这个初始应力后，其切应力才与速度梯度成正比，即

$$\tau=\tau_0+\mu\frac{\mathrm{d}u}{\mathrm{d}y} \tag{1-6}$$

假塑性流体，如泥浆等，其切应力与速度梯度的关系是

$$\tau=\mu\left(\frac{\mathrm{d}u}{\mathrm{d}y}\right)^n,\quad n<1 \tag{1-7}$$

胀塑性流体，如乳化液等，其切应力与速度梯度的关系是

$$\tau=\mu\left(\frac{\mathrm{d}u}{\mathrm{d}y}\right)^n,\quad n>1 \tag{1-8}$$

1.1.3 流体力学中的力与压强

1. 质量力

与流体微团质量大小有关并且集中在微团质量中心的力称为质量力(body force)。在重力场中有重力 mg；直线运动时，有惯性力 ma。质量力是一个矢量，一般用单位质量所具

有的质量力来表示，其形式如下：

$$f = f_x i + f_y j + f_z k \tag{1-9}$$

式中：f_x，f_y，f_z 为单位质量力在各轴上的投影。

2. 表面力

大小与表面面积有关而且分布作用在流体表面上的力称为表面力(surface force)。表面力按其作用方向可以分为两种：一是沿表面内法线方向的压力，称为正压力；另一种是沿表面切向的摩擦力，称为切向力。

对于理想流体的流动，流体质点只受到正压力，没有切向力；对于粘性流体的流动，流体质点所受到的作用力既有正压力，也有切向力。

作用在静止流体上的表面力只有沿表面内法线方向的正压力。单位面积上所受到的表面力称为这一点处的静压强。静压强具有两个特征：①静压强的方向垂直指向作用面；②流场内一点处静压强的大小与方向无关。

3. 表面张力

在液体表面，界面上液体间的相互作用力称为张力。在液体表面有自动收缩的趋势，收缩的液面存在相互作用的与该处液面相切的拉力，称为液体的表面张力(surface tension)。正是这种力的存在，引起弯曲液面内外出现压强差以及常见的毛细现象等。

试验表明，表面张力大小与液面的截线长度 L 成正比，即

$$T = \sigma L \tag{1-10}$$

式中：σ 为表面张力系数，它表示液面上单位长度截线上的表面张力，其大小由物质种类决定，其单位为 N/m。

4. 绝对压强、相对压强及真空度

标准大气压的压强是 101325Pa(760mm 汞柱)，通常用 p_{atm} 表示。若压强大于大气压，则以该压强为计算基准得到的压强称为相对压强(relative pressure)，也称为表压强，通常用 p_r 表示。若压强小于大气压，则压强低于大气压的值就称为真空度(vacuum)，通常用 p_v 表示。如以压强 0Pa 为计算的基准，则这个压强就称为绝对压强(absolute pressure)，通常用 p_s 表示。这三者的关系如下：

$$p_r = p_s - p_{atm} \tag{1-11}$$

$$p_v = p_{atm} - p_s \tag{1-12}$$

在流体力学中，压强都用符号 p 表示，但一般来说有一个约定：对于液体，压强用相对压强；对于气体，特别是马赫数大于 0.1 的流动，应视为可压缩流，压强用绝对压强。

压强的单位较多，一般用 Pa，也可用 bar，还可以用汞柱、水柱，这些单位换算如下：

$$1Pa=1N/m^2$$

$$1bar=105Pa$$

$$1p_{atm}=760mmHg=10.33mH_2O=101325Pa$$

5. 静压、动压和总压

对于静止状态下的流体，只有静压强。对于流动状态的流体，有静压强(static pressure)、

动压强(dynamic pressure)、测压管压强(manometric tube pressure)和总压强(total pressure)之分。下面从伯努利(Bernoulli)方程(也有人称其为伯努里方程)中分析它们的意义。

伯努利方程阐述一条流线上流体质点的机械能守恒，对于理想流体的不可压缩流动其表达式如下：

$$\frac{p}{\rho g}+\frac{v^2}{2g}+z=H \tag{1-13}$$

式中：$p/\rho g$ 称为压强水头，也是压能项，为静压强；$v^2/2g$ 称为速度水头，也是动能项；z 称为位置水头，也是重力势能项，这三项之和就是流体质点的总的机械能；H 称为总的水头高。

将式(1-13)两边同时乘以 ρg，则有

$$p+\frac{1}{2}\rho v^2+\rho gz=\rho gH \tag{1-14}$$

式中：p 称为静压强，简称静压；$\frac{1}{2}\rho v^2$ 称为动压强，简称动压；ρgH 称为总压强，简称总压。对于不考虑重力的流动，总压就是静压和动压之和。

1.1.4 流体运动的描述

1. 流体运动描述的方法

描述流体物理量有两种方法，一种是拉格朗日描述；一种是欧拉描述。

拉格朗日(Lagrange)描述也称随体描述，它着眼于流体质点，并将流体质点的物理量认为是随流体质点及时间变化的，即把流体质点的物理量表示为拉格朗日坐标及时间的函数。设拉格朗日坐标为(a,b,c)，以此坐标表示的流体质点的物理量，如矢径、速度、压强等等在任一时刻 t 的值，便可以写为 a、b、c 及 t 的函数。

若以 f 表示流体质点的某一物理量，其拉格朗日描述的数学表达式为

$$f=f(a,b,c,t) \tag{1-15}$$

例如，设时刻 t 流体质点的矢径即 t 时刻流体质点的位置以 r 表示，其拉格朗日描述为

$$r=r(a,b,c,t) \tag{1-16}$$

同样，质点的速度的拉格朗日描述是

$$v=v(a,b,c,t) \tag{1-17}$$

欧拉描述，也称空间描述，它着眼于空间点，认为流体的物理量随空间点及时间而变化，即把流体物理量表示为欧拉坐标及时间的函数。设欧拉坐标为(q_1,q_2,q_3)，用欧拉坐标表示的各空间点上的流体物理量如速度、压强等，在任一时刻 t 的值，可写为 q_1、q_2、q_3 及 t 的函数。从数学分析知道，当某时刻一个物理量在空间的分布一旦确定，该物理量在此空间形成一个场。因此，欧拉描述实际上描述了一个个物理量的场。

若以 f 表示流体的一个物理量，其欧拉描述的数学表达式是(设空间坐标取用直角坐标)

$$f=F(x,y,z,t)=F(r,t) \tag{1-18}$$

如流体速度的欧拉描述是

$$v=v(x,y,z,t) \tag{1-19}$$

2. 拉格朗日描述与欧拉描述之间的关系

拉格朗日描述着眼于流体质点，将物理量视为流体坐标与时间的函数；欧拉描述着眼于空间点，将物理量视为空间坐标与时间的函数。它们可以描述同一物理量，必定互相相关。设表达式 $f=f(a,b,c,t)$ 表示流体质点(a,b,c)在 t 时刻的物理量；表达式 $f=F(x,y,z,t)$ 表示空间点(x,y,z)在时刻 t 的同一物理量。如果流体质点(a,b,c)在 t 时刻恰好运动到空间点(x,y,z)上，则应有

$$\begin{cases} x = x(a,b,c,t) \\ y = y(a,b,c,t) \\ z = z(a,b,c,t) \end{cases} \tag{1-20}$$

$$F(x,y,z,t) = f(a,b,c,t) \tag{1-21}$$

事实上，将式(1-16)代入式(1-21)左端，即有

$$\begin{aligned} F(x,y,z,t) &= F[x(a,b,c,t),y(a,b,c,t),z(a,b,c,t),t] \\ &= f(a,b,c,t) \end{aligned} \tag{1-22}$$

或者反解式(1-16)，得到

$$\begin{cases} a = a(x,y,z,t) \\ b = b(x,y,z,t) \\ c = c(x,y,z,t) \end{cases} \tag{1-23}$$

将式(1-23)代入式(1-21)的右端，也应有

$$\begin{aligned} f(a,b,c,t) &= f[a(x,y,z,t),b(x,y,z,t),c(x,y,z,t),t] \\ &= F(x,y,z,t) \end{aligned} \tag{1-24}$$

由此，可以通过拉格朗日描述推出欧拉描述，同样也可以由欧拉描述推出拉格朗日描述。

3. 随体导数

流体质点物理量随时间的变化率称为随体导数(substantial derivative)，或物质导数、质点导数。

按拉格朗日描述，物理量 f 表示为 $f=f(a,b,c,t)$，f 的随体导数就是跟随质点(a,b,c)的物理量 f 对时间 t 的导数 $\partial f/\partial t$。例如，速度 $v(a,b,c,t)$ 是矢径 $r(a,b,c,t)$ 对时间的偏导数，

$$v(a,b,c,t) = \frac{\partial r(a,b,c,t)}{\partial t} \tag{1-25}$$

即随体导数就是偏导数。

按欧拉描述，物理量 f 表示为 $f=F(x,y,z,t)$，但 $\partial F/\partial t$ 并不表示随体导数，它只表示物理量在空间点 (x,y,z,t) 上的时间变化率。而随体导数必须跟随 t 时刻位于 (x,y,z,t) 空间点上的那个流体质点，其物理量 f 的时间变化率。由于该流体质点是运动的，即 x、y、z 是变的，若以 a、b、c 表示该流体质点的拉格朗日坐标，则 x、y、z 将依式(1-16)变化，从而 $f=F(x,y,z,t)$的变化依连锁法则处理。因此，物理量 $f=F(x,y,z,t)$的随体导数是

$$\begin{aligned}\frac{\mathrm{D}F(x,y,z,t)}{\mathrm{D}t} &= \mathrm{D}F[x(a,b,c,t),y(a,b,c,t),z(a,b,c,t),t]\\ &= \frac{\partial F}{\partial x}\frac{\partial x}{\partial t}+\frac{\partial F}{\partial y}\frac{\partial y}{\partial t}+\frac{\partial F}{\partial z}\frac{\partial z}{\partial t}+\frac{\partial F}{\partial t}\\ &= \frac{\partial F}{\partial x}u+\frac{\partial F}{\partial y}v+\frac{\partial F}{\partial z}w+\frac{\partial F}{\partial t}\\ &= (v\cdot\nabla)F+\frac{\partial F}{\partial t}\end{aligned} \tag{1-26}$$

式中：$\mathrm{D}/\mathrm{D}t$ 表示随体导数。

从中可以看出，对于质点物理量的随体导数，欧拉描述与拉格朗日描述大不相同。前者是两者之和，而后者是直接的偏导数。

4. 定常流动与非定常流动

根据流体流动过程以及流动过程中的流体的物理参数是否与时间相关，可将流动分为定常流动(steady flow)与非定常流动(unsteady flow)。

定常流动：流体流动过程中各物理量均与时间无关，这种流动称为定常流动。

非定常流动：流体流动过程中某个或某些物理量与时间有关，则这种流动称为非定常流动。

5. 流线与迹线

常用流线和迹线来描述流体的流动。

迹线(track)：随着时间的变化，空间某一点处的流体质点在流动过程中所留下的痕迹称为迹线。在 t=0 时刻，位于空间坐标(a,b,c)处的流体质点，其迹线方程为

$$\begin{cases}\mathrm{d}x(a,b,c,t)=u\mathrm{d}t\\ \mathrm{d}y(a,b,c,t)=v\mathrm{d}t\\ \mathrm{d}z(a,b,c,t)=w\mathrm{d}t\end{cases} \tag{1-27}$$

式中：u、v、w 分别为流体质点速度的三个分量；x、y、z 为在 t 时刻此流体质点的空间位置。

流线(streamline)：在同一个时刻，由不同的无数多个流体质点组成的一条曲线，曲线上每一点处的切线与该质点处流体质点的运动方向平行。流场在某一时刻 t 的流线方程为

$$\frac{\mathrm{d}x}{u(x,y,z,t)}=\frac{\mathrm{d}y}{v(x,y,z,t)}=\frac{\mathrm{d}z}{w(x,y,z,t)} \tag{1-28}$$

对于定常流动，流线的形状不随时间变化，而且流体质点的迹线与流线重合。在实际流场中除驻点或奇点外，流线不能相交，不能突然转折。

6. 流量与净通量

流量(flux)：单位时间内流过某一控制面的流体体积称为该控制面的流量 Q，其单位为 $\mathrm{m^3/s}$。若单位时间内流过的流体是以质量计算，则称为质量流量 Q_m；不加说明时“流量”一词概指体积流量。在曲面控制面上有

$$Q=\iint_A v\cdot n\mathrm{d}A \tag{1-29}$$

净通量(net flux)：在流场中取整个封闭曲面作为控制面 A，封闭曲面内的空间称为控制体。流体经一部分控制面流入控制体，同时也有流体经另一部分控制面从控制体中流出，此时流出的流体减去流入的流体，所得出的流量称为流过全部封闭控制面 A 的净流量(或净通量)，通过式(1-30)计算：

$$q = \oiint_A v \cdot n\mathrm{d}A \tag{1-30}$$

对于不可压缩流体来说，流过任意封闭控制面的净通量等于 0。

7. 有旋流动与有势流动

由速度分解定理，流体质点的运动可以分解为：

(1)　随同其他质点的平动；

(2)　自身的旋转运动；

(3)　自身的变形运动(拉伸变形和剪切变形)。

在流动过程中，若流体质点自身做无旋运动(irrotational flow)，则称流动是无旋的，也就是有势的，否则就称流动是有旋流动(rotational flow)。流体质点的旋度是一个矢量，通常用 $\boldsymbol{\omega}$ 表示，其大小为

$$\omega = \frac{1}{2}\begin{vmatrix} i & j & k \\ \dfrac{\partial}{\partial x} & \dfrac{\partial}{\partial y} & \dfrac{\partial}{\partial z} \\ u & v & w \end{vmatrix} \tag{1-31}$$

若 $\omega=0$，则称流动为无旋流动，否则就是有旋流动。

$\boldsymbol{\omega}$ 与流体的流线或迹线形状无关；粘性流动一般为有旋流动；对于无旋流动，伯努利方程适用于流场中任意两点之间；无旋流动也称为有势流动(potential flow)，即存在一个势函数 $\varphi(x,y,z,t)$，满足：

$$V = \mathrm{grad}\varphi \tag{1-32}$$

即

$$u = \frac{\partial\varphi}{\partial x},\ v = \frac{\partial\varphi}{\partial y},\ w = \frac{\partial\varphi}{\partial z} \tag{1-33}$$

8. 层流与湍流

流体的流动分为层流流动(laminar flow)和湍流流动(turbulent flow)。从试验的角度来看，层流流动就是流体层与层之间相互没有任何干扰，层与层之间既没有质量的传递也没有动量的传递；而湍流流动中层与层之间相互有干扰，而且干扰的力度还会随着流动而加大，层与层之间既有质量的传递又有动量的传递。

判断流动是层流还是湍流，是看其雷诺数是否超过临界雷诺数。雷诺数的定义如下：

$$Re = \frac{VL}{\nu} \tag{1-34}$$

式中：V 为截面的平均速度；L 为特征长度；ν 为流体的运动粘度。

对于圆形管内流动，特征长度 L 取圆管的直径 d。一般认为临界雷诺数为 2320，即

$$Re=\frac{vd}{\nu} \tag{1-35}$$

当 $Re<2320$ 时，管中是层流；当 $Re>2320$ 时，管中是湍流。

对于异型管道内的流动，特征长度取水力直径 d_{H}，则雷诺数的表达式为

$$Re=\frac{Vd_{\mathrm{H}}}{\nu} \tag{1-36}$$

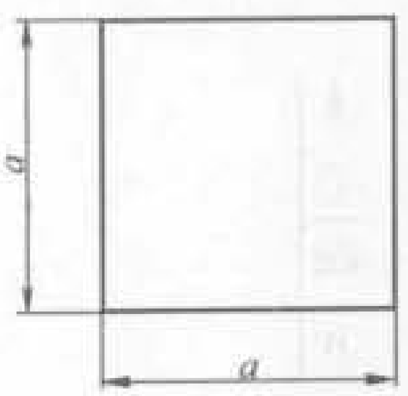
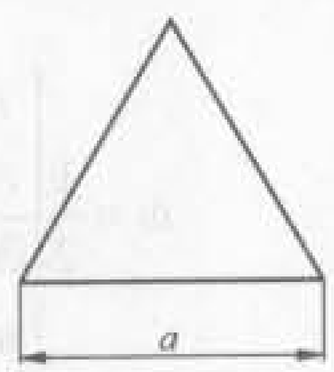
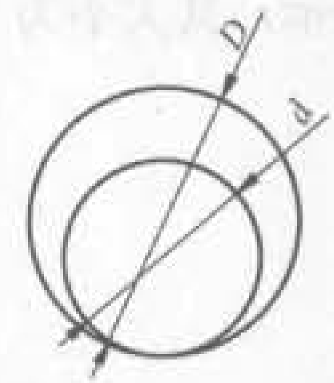
异型管道水力直径的定义如下：

$$d_{\mathrm{H}}=4\frac{A}{S} \tag{1-37}$$

式中：A 为过流断面的面积；S 为过流断面上流体与固体接触的周长。

临界雷诺数根据形状的不同而有所差别。根据试验几种异型管道的临界雷诺数如表 1-1 所示。

表 1-1　几种异型管道的临界雷诺数

	正方形	正三角形	偏心缝隙
管道截面形状			
$Re=\frac{Vd_{\mathrm{H}}}{\nu}$	$\frac{Va}{\nu}$	$\frac{Va}{\sqrt{3}\nu}$	$\frac{V}{\nu}(D-d)$
Re_c	2070	1930	1000

对于平板的外部绕流，特征长度取沿流动方向的长度，其临界雷诺数为 $5\times10^5\sim3\times10^6$。

1.2　CFD 基本模型

流体流动所遵循的物理定律，是建立流体运动基本方程组的依据。这些定律主要包括质量守恒、动量守恒、动量矩守恒、能量守恒、热力学第二定律，加上状态方程、本构方程。在实际计算时，还要考虑不同的流态，如层流与湍流。

1.2.1　基本控制方程

1. 系统与控制体

在流体力学中，系统是指某一确定流体质点集合的总体。系统以外的环境称为外界。分隔系统与外界的界面，称为系统的边界。系统通常是研究的对象，外界则用来区别于系统。系统将随系统内质点一起运动，系统内的质点始终包含在系统内，系统边界的形状和所围空间的大小可随运动而变化。系统与外界无质量交换，但可以有力的相互作用，及能

量(热和功)交换。

控制体是指在流体所在的空间中，以假想或真实流体边界包围，固定不动形状任意的空间体积。包围这个空间体积的边界面，称为控制面。控制体的形状与大小不变，并相对于某坐标系固定不动。控制体内的流体质点组成并非不变的。控制体既可通过控制面与外界有质量和能量交换，也可与控制体外的环境有力的相互作用。

2. 质量守恒方程(连续性方程)

在流场中，流体通过控制面 A_1 流入控制体，同时也会通过另一部分控制面 A_2 流出控制体，在这期间控制体内部的流体质量也会发生变化。按照质量守恒定律，流入的质量与流出的质量之差，应该等于控制体内部流体质量的增量，由此可导出流体流动连续性方程的积分形式为

$$\frac{\partial}{\partial t}\iiint_V \rho \mathrm{d}x\mathrm{d}y\mathrm{d}z + \oiint_A \rho v \cdot n \mathrm{d}A = 0 \tag{1-38}$$

式中：V 表示控制体，A 表示控制面。等式左边第一项表示控制体 V 内部质量的增量；第二项表示通过控制表面流入控制体的净通量。

根据数学中的奥-高公式，在直角坐标系下可将其化为微分形式：

$$\frac{\partial \rho}{\partial t} + u\frac{\partial(\rho u)}{\partial x} + v\frac{\partial(\rho v)}{\partial y} + w\frac{\partial(\rho w)}{\partial z} = 0 \tag{1-39}$$

对于不可压缩均质流体，密度为常数，则有

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{1-40}$$

对于圆柱坐标系，其形式为

$$\frac{\partial \rho}{\partial t} + \frac{\rho v_r}{r} + \frac{\partial(\rho v_r)}{\partial r} + \frac{\partial(\rho v_\theta)}{r\partial \theta} + \frac{\partial(\rho v_z)}{\partial z} = 0 \tag{1-41}$$

对于不可压缩均质流体，密度为常数，则有

$$\frac{v_r}{r} + \frac{\partial v_r}{\partial r} + \frac{\partial v_\theta}{r\partial \theta} + \frac{\partial v_z}{\partial z} = 0 \tag{1-42}$$

3. 动量守恒方程(运动方程)

动量守恒是流体运动时应遵循的另一个普遍定律，描述为：在一给定的流体系统，其动量的时间变化率等于作用于其上的外力总和，其数学表达式即为动量守恒方程，也称为运动方程，或 N-S 方程，其微分形式表达如下：

$$\begin{cases} \rho \dfrac{\mathrm{d}u}{\mathrm{d}t} = \rho F_{bx} + \dfrac{\partial p_{xx}}{\partial x} + \dfrac{\partial p_{yx}}{\partial y} + \dfrac{\partial p_{zx}}{\partial z} \\ \rho \dfrac{\mathrm{d}v}{\mathrm{d}t} = \rho F_{by} + \dfrac{\partial p_{xy}}{\partial x} + \dfrac{\partial p_{yy}}{\partial y} + \dfrac{\partial p_{zy}}{\partial z} \\ \rho \dfrac{\mathrm{d}w}{\mathrm{d}t} = \rho F_{bz} + \dfrac{\partial p_{xz}}{\partial x} + \dfrac{\partial p_{yz}}{\partial y} + \dfrac{\partial p_{zz}}{\partial z} \end{cases} \tag{1-43}$$

式中：F_{bx}、F_{by}、F_{bz} 分别是单位质量流体上的质量力在三个方向上的分量；p_{yx} 是流体内应力张量的分量。

动量守恒方程在实际应用中有许多表达形式，其中比较常见的有如下几种。

(1) 可压缩粘性流体的动量守恒方程

$$\begin{cases} \rho\dfrac{\mathrm{d}u}{\mathrm{d}t}=\rho f_x+\dfrac{\partial p}{\partial x}+\dfrac{\partial}{\partial x}\left\{\mu\left[2\dfrac{\partial u}{\partial x}-\dfrac{2}{3}\left(\dfrac{\partial u}{\partial x}+\dfrac{\partial v}{\partial y}+\dfrac{\partial w}{\partial z}\right)\right]\right\}+ \\ \qquad \dfrac{\partial}{\partial y}\left[\mu\left(\dfrac{\partial u}{\partial y}+\dfrac{\partial v}{\partial x}\right)\right]+\dfrac{\partial}{\partial z}\left[\mu\left(\dfrac{\partial w}{\partial x}+\dfrac{\partial u}{\partial z}\right)\right] \\ \rho\dfrac{\mathrm{d}v}{\mathrm{d}t}=\rho f_y+\dfrac{\partial p}{\partial y}+\dfrac{\partial}{\partial y}\left\{\mu\left[2\dfrac{\partial v}{\partial y}-\dfrac{2}{3}\left(\dfrac{\partial u}{\partial x}+\dfrac{\partial v}{\partial y}+\dfrac{\partial w}{\partial z}\right)\right]\right\}+ \\ \qquad \dfrac{\partial}{\partial z}\left[\mu\left(\dfrac{\partial v}{\partial z}+\dfrac{\partial w}{\partial y}\right)\right]+\dfrac{\partial}{\partial x}\left[\mu\left(\dfrac{\partial u}{\partial y}+\dfrac{\partial v}{\partial x}\right)\right] \\ \rho\dfrac{\mathrm{d}w}{\mathrm{d}t}=\rho f_z+\dfrac{\partial p}{\partial z}+\dfrac{\partial}{\partial z}\left\{\mu\left[2\dfrac{\partial w}{\partial z}-\dfrac{2}{3}\left(\dfrac{\partial u}{\partial x}+\dfrac{\partial v}{\partial y}+\dfrac{\partial w}{\partial z}\right)\right]\right\}+ \\ \qquad \dfrac{\partial}{\partial x}\left[\mu\left(\dfrac{\partial w}{\partial x}+\dfrac{\partial u}{\partial z}\right)\right]+\dfrac{\partial}{\partial z}\left[\mu\left(\dfrac{\partial v}{\partial z}+\dfrac{\partial w}{\partial z}y\right)\right] \end{cases} \tag{1-44}$$

(2) 常粘性流体的动量守恒方程

$$\rho\frac{\mathrm{d}v}{\mathrm{d}t}=\rho F-\mathrm{grad}p+\frac{\mu}{3}\mathrm{grad}(\mathrm{div}v)+\mu\nabla^2 v \tag{1-45}$$

(3) 常密度常粘性流体的动量守恒方程

$$\rho\frac{\mathrm{d}v}{\mathrm{d}t}=\rho F-\mathrm{grad}p+\mu\nabla^2 v \tag{1-46}$$

(4) 无粘性流体的动量守恒方程(欧拉方程)

$$\rho\frac{\mathrm{d}v}{\mathrm{d}t}=\rho F-\mathrm{grad}p \tag{1-47}$$

(5) 静力学方程

$$\rho F=\mathrm{grad}p \tag{1-48}$$

(6) 相对运动方程

在非惯性参考系中的相对运动方程是研究像大气、海洋及旋转系统中流体运动的所必须考虑的。由理论力学得知，绝对速度 v_a 为相对速度 v_r 及牵连速度 v_e 之和，即

$$v_a=v_r+v_e \tag{1-49}$$

其中，$v_e=v_0+\Omega\times r$，v_0 为运动系中的平动速度，Ω 是其转动角速度，r 为质点矢径。

而绝对加速度 a_a 为相对加速度 a_r、牵连加速度 a_e 及科氏加速度 a_c 之和，即

$$a_a=a_r+a_e+a_c \tag{1-50}$$

其中，$a_e=\dfrac{\mathrm{d}v_0}{\mathrm{d}t}+\dfrac{\mathrm{d}\Omega}{\mathrm{d}t}\times r+\Omega\times(\Omega\times r)$，$a_c=2\Omega\times v_r$。

将绝对加速度代入运动方程，即得到流体的相对运动方程

$$\rho\frac{\mathrm{d}v_r}{\mathrm{d}t}=\rho F_b+\mathrm{div}P-a_e-2\Omega v_r \tag{1-51}$$

4. 能量守恒方程

将热力学第一定律应用于流体运动，把式(1-51)各项用有关的流体物理量表示出来，即是能量方程。如式(1-52)所示。

$$\frac{\partial}{\partial t}(\rho E)+\frac{\partial}{\partial x_i}[u_i(\rho E+p)]=\frac{\partial}{\partial x_i}\left[k_{\text{eff}}\frac{\partial T}{\partial x_i}-\sum_{j'}h_{j'}J_{j'}+u_j(\tau_{ij})_{\text{eff}}\right]+S_h \tag{1-52}$$

式中：$E=h-\frac{p}{\rho}+\frac{u_i^2}{2}$；$k_{\text{eff}}$是有效热传导系数，$k_{\text{eff}}=k+k_t$，其中$k_t$是湍流热传导系数，根据所使用的湍流模型来定义；$J_{j'}$是组分$j$的扩散流量；$S_h$包括了化学反应热以及其他用户定义的体积热源项；方程右边的前3项分别描述了热传导、组分扩散和粘性耗散带来的能量输运。

1.2.2 湍流模型

湍流是自然界广泛存在的流动现象。大气、海洋环境的流动，飞行器和船舰的绕流，叶轮机械、化学反应器、核反应器中的流体运动都是湍流。湍流流动的核心特征是其在物理上近乎于无穷多的尺度和数学上强烈的非线性，这使得人们无论是通过理论分析、实验研究还是计算机模拟来彻底认识湍流都非常困难。回顾计算流体力学的发展，特别是活跃的20世纪80年代，不仅提出和发展了一大批高精度、高分辨率的计算格式，从主控方程看相当成功地解决了欧拉方程的数值模拟，可以说欧拉方程数值模拟方法的精度已接近于它有效使用范围的极限；同时还发展了一大批有效的网格生成技术及相应的软件，具体实现了工程计算所需要的复杂外形的计算网格；且随着计算机的发展，无论从计算时间还是从计算费用考虑，欧拉方程都已能适用于各种实践所需。在此基础上，20世纪80年代还进行了求解可压缩雷诺平均方程及其三维定态粘流流动的模拟。20世纪90年代又开始一个非定常粘流流场模拟的新局面，这里所说的粘流流场具有高雷诺数、非定常、不稳定、剧烈分离流动的特点，显然需要继续探求更高精度的计算方法和更实用可靠的网格生成技术。但更为重要的关键性的决策将是，研究湍流机理，建立相应的模式，并进行适当的模拟仍是解决湍流问题的重要途径。

1. 湍流模型分类

湍流流动模型很多，但大致可以归纳为以下3类。

第一类是湍流输运系数模型，即将速度脉动的二阶关联量表示成平均速度梯度与湍流粘性系数的乘积，用笛卡儿张量表示为

$$-\rho\overline{u_i'u_j'}=\mu_t\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)-\frac{2}{3}\rho k\delta_{ij} \tag{1-53}$$

模型的任务就是给出计算湍流粘性系数μ_t的方法。根据建立模型所需要的微分方程的数目，可以分为零方程模型(代数方程模型)、单方程模型和双方程模型。

第二类是抛弃了湍流输运系数的概念，直接建立湍流应力和其他二阶关联量的输运方程。

第三类是大涡模拟。前两类是以湍流的统计结构为基础，对所有涡旋进行统计平均。

大涡模拟把湍流分成大尺度湍流和小尺度湍流，通过求解三维经过修正的 Navier-Stokes 方程(纳维−斯托克斯方程，简称 N-S 方程)，得到大涡旋的运动特性，而对小涡旋运动还采用上述的模型。

实际求解中，选用什么模型要根据具体问题的特点来决定。选择的一般原则是精度要高，应用简单，节省计算时间，同时也具有通用性。

Fluent 提供的湍流模型包括：单方程(Spalart-Allmaras)模型、双方程模型(标准 k-ε 模型、重整化群 k-ε 模型、可实现 k-ε 模型)及雷诺应力模型和大涡模拟，如图 1-1 所示。

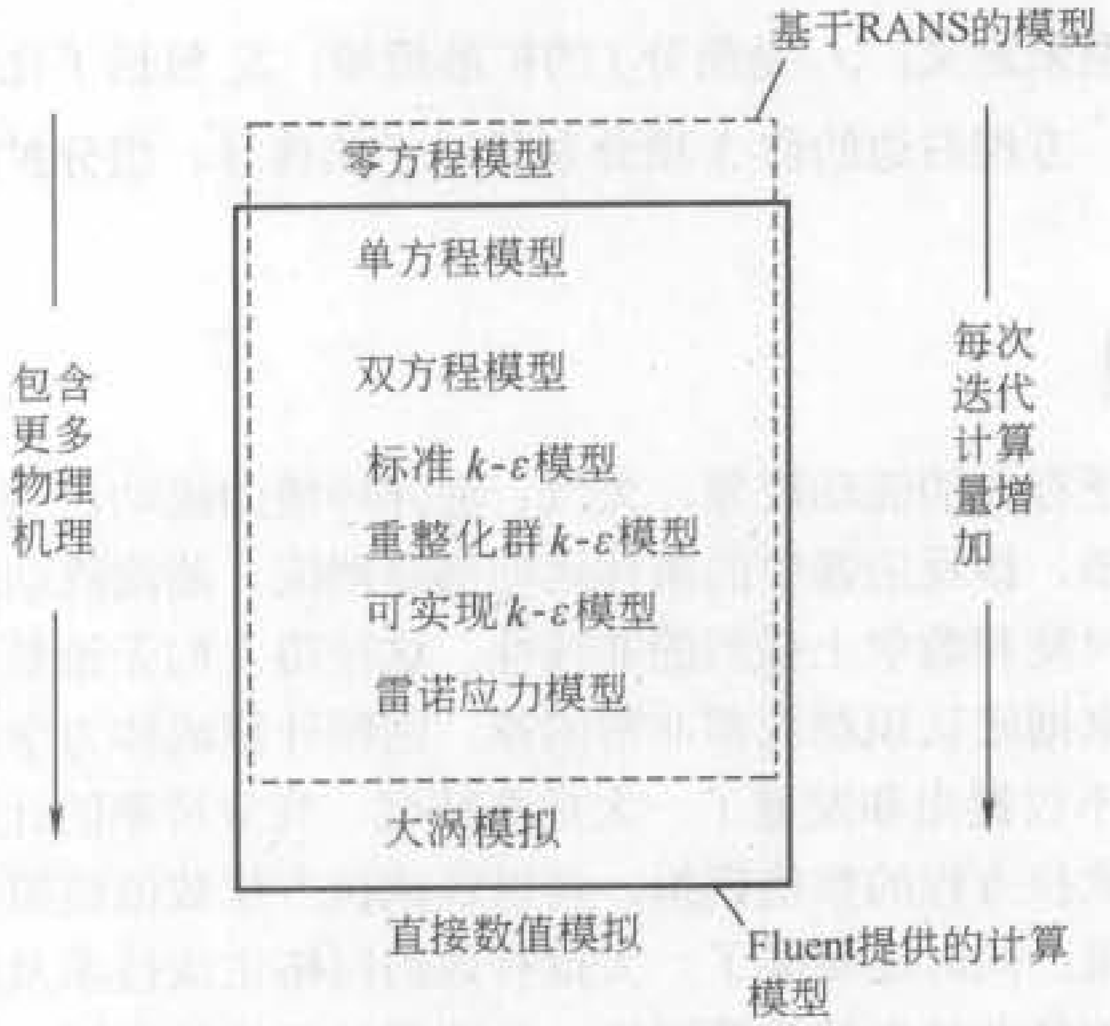


图 1-1　湍流模型详解

2. 平均量输运方程

雷诺平均就是把 Navier-Stokes 方程中的瞬时变量分解成平均量和脉动量两部分。对于速度，有

$$u_i = \overline{u_i} + u_i' \tag{1-54}$$

式中：$\overline{u_i}$ 和 u_i' 分别是平均速度和脉动速度($i=1,2,3$)。

类似地，对于压力等其他标量，也有

$$\phi = \overline{\phi} + \phi' \tag{1-55}$$

式中：ϕ 表示标量，如压力、能量、组分浓度等。

把上面的表达式代入瞬时的连续与动量方程，并取平均(去掉平均速度上的横线)，可以把连续与动量方程写成如下的笛卡儿坐标系下的张量形式：

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i) = 0 \tag{1-56}$$

$$\rho \frac{\mathrm{d}u_i}{\mathrm{d}t} = \frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial u_l}{\partial x_l}\right)\right] + \frac{\partial}{\partial x_j}\left(-\rho\overline{u_i' u_j'}\right) \tag{1-57}$$

上面两个方程称为雷诺平均的 Navier-Stokes(RANS)方程。它们和瞬时 Navier-Stokes 方

程有相同的形式，只是速度或其他求解变量变成了时间平均量。额外多出来的项 $-\rho\overline{u_i'u_j'}$ 是雷诺应力，表示湍流的影响。

对于密度变化的流动过程，如燃烧问题，需要采用法夫雷(Favre)平均才可以求解。法夫雷平均就是除了压力和密度本身以外，所有变量都用密度加权平均。变量的密度加权平均定义如下：

$$\tilde{\phi}=\overline{\rho\phi}/\overline{\rho} \tag{1-58}$$

式中：符号~表示密度加权平均，对应于密度加权平均值的脉动值用 ϕ'' 表示，有 $\phi=\tilde{\phi}+\phi''$。显然，这种脉动值的简单平均值不为零，但它的密度加权平均值等于零，即 $\overline{\phi''}\neq 0$，$\overline{\rho\phi''}=0$。

为了求解方程(1-57)，必须模拟雷诺应力项以使方程封闭。通常的方法是应用 Boussinesq 假设，认为雷诺应力与平均速度梯度成正比，表达式如下：

$$-\rho\overline{u_i'u_j'}=\mu_t\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)-\frac{2}{3}\left(\rho k+\mu_t\frac{\partial u_i}{\partial x_i}\right)\delta_{ij} \tag{1-59}$$

Boussinesq 假设被用于单方程模型和 k-ε 双方程模型。这种近似方法好处是与求解湍流粘性系数有关的计算时间比较少。例如，在 Spalart-Allmaras 单方程模型中只多求解一个表示湍流粘性的输运方程；在 k-ε 双方程模型中只需多求解湍动能 k 和耗散率 ε 两个方程，湍流粘性系数用湍动能 k 和耗散率 ε 的函数来描述。Boussinesq 假设的不足之处是假设 μ_t 是个各向同性标量，对于一些复杂流动，该条件并不是严格成立，所以具有其应用局限性。

另外的近似方法是求解雷诺应力各分量的输运方程。这也需要额外再求解一个标量方程，通常是耗散率 ε 方程。这就意味着对于二维湍流流动问题，需要多求解 4 个输运方程，而三维湍流问题需要多求解 7 个方程，需要较多的计算时间，要求更高的计算机内存。

在很多情况下基于 Boussinesq 假设的模型很好用，而且计算量并不是很大。但是，如果湍流场各向异性很明显，如强旋流动以及应力取得的二次流等流动中，求解 RSM 模型可以得到更好的结果。

3. 常用湍流模型简介

1)　单方程(Spalart-Allmaras)模型

单方程模型求解变量是 $\tilde{v}$，表征出了近壁(粘性影响)区域以外的湍流运动粘性系数。$\tilde{v}$ 的输运方程为

$$\rho\frac{\mathrm{d}\tilde{v}}{\mathrm{d}t}=G_v+\frac{1}{\sigma_{\tilde{v}}}\left[\frac{\partial}{\partial x_j}\left\{(\mu+\rho\tilde{v})\frac{\partial\tilde{v}}{\partial x_j}\right\}+C_{b2}\left(\frac{\partial\tilde{v}}{\partial x_j}\right)\right]-Y_v \tag{1-60}$$

式中：G_v 是湍流粘性产生项；Y_v 是由于壁面阻挡与粘性阻尼引起的湍流粘性的减少；$\sigma_{\tilde{v}}$ 和 C_{b2} 是常数；v 是分子运动粘性系数。

湍流粘性系数 $\mu_t=\rho\tilde{v}f_{v1}$，其中，$f_{v1}$ 是粘性阻尼函数，定义为 $f_{v1}=\dfrac{\chi^3}{\chi^3+C_{v1}{}^3}$，$\chi\equiv\dfrac{\tilde{v}}{v}$。

而湍流粘性产生项 G_v 模拟为 $G_v=C_{b1}\rho\tilde{S}\tilde{v}$，其中 $\tilde{S}\equiv S+\dfrac{\tilde{v}}{k^2d^2}f_{v2}$，$f_{v2}=1-\dfrac{\chi}{1+\chi f_{v1}}$，$C_{b1}$ 和 k

是常数，d 是计算点到壁面的距离；$S \equiv \sqrt{2\Omega_{ij}\Omega_{ij}}$，$\Omega_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_i} - \frac{\partial u_j}{\partial x_j}\right)$。在 Fluent 软件中，考虑到平均应变率对湍流产生也起到很大作用，$S \equiv |\Omega_{ij}| + C_{\text{prod}}\min(0, |S_{ij}| - |\Omega_{ij}|)$，其中，$C_{\text{prod}}=2.0$，$|\Omega_{ij}| \equiv \sqrt{2\Omega_{ij}\Omega_{ij}}$，$|S_{ij}| \equiv \sqrt{2S_{ij}S_{ij}}$，平均应变率 $S_{ij} = \frac{1}{2}\left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j}\right)$。

在涡量超过应变率的计算区域计算出来的涡旋粘性系数变小。这适合涡流靠近涡旋中心的区域，那里只有"单纯"的旋转，湍流受到抑止。包含应变张量的影响更能体现旋转对湍流的影响。忽略了平均应变，估计的涡旋粘性系数产生项偏高。

湍流粘性系数减少项 Y_v 为 $Y_v = C_{w1}\rho f_w \left(\frac{\tilde{v}}{d}\right)^2$，其中，$f_w = g\left(\frac{1+C_{w3}^6}{g_6 + C_{w3}^6}\right)^{1/6}$，$g = r + C_{w2}(r^6 - r)$，$r \equiv \frac{\tilde{v}}{\tilde{S}k^2d^2}$，$C_{w1}$、$C_{w2}$、$C_{w3}$ 是常数，在计算 r 时用到的 $\tilde{S}$ 受平均应变率的影响。

上面的模型常数在 Fluent 软件中默认值为 $C_{b1} = 0.1335$，$C_{b2} = 0.622$，$\sigma_{\tilde{v}} = 2/3$，$C_{v1} = 7.1$，$C_{w1} = C_{b1}/k^2 + (1+C_{b2})/\sigma_{\tilde{v}}$，$C_{w2} = 0.3$，$C_{w3} = 2.0$，$k = 0.41$。

2) 标准 k-ε 模型

标准 k-ε 模型需要求解湍动能及其耗散率方程。湍动能输运方程是通过精确的方程推导得到的，但耗散率方程是通过物理推理，数学上模拟相似原形方程得到的。该模型假设流动为完全湍流，分子粘性的影响可以忽略。因此，标准 k-ε 模型只适合完全湍流的流动过程模拟。标准 k-ε 模型的湍动能 k 和耗散率 ε 方程为如下形式：

$$\rho \frac{\mathrm{d}k}{\mathrm{d}t} = \frac{\partial}{\partial x_i}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_i}\right] + G_k + G_b - \rho\varepsilon - Y_M \tag{1-61}$$

$$\rho \frac{\mathrm{d}\varepsilon}{\mathrm{d}t} = \frac{\partial}{\partial x_i}\left[\left(\mu + \frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_i}\right] + C_{1\varepsilon}\frac{\varepsilon}{k}(G_k + C_{3\varepsilon}G_b) - C_{2\varepsilon}\rho\frac{\varepsilon^2}{k} \tag{1-62}$$

式中：G_k 表示由于平均速度梯度引起的湍动能产生，G_b 表示由于浮力影响引起的湍动能产生；Y_M 表示可压缩湍流脉动膨胀对总的耗散率的影响。湍流粘性系数 $\mu_t = \rho C_\mu \frac{k^2}{\varepsilon}$。

在 Fluent 中，作为默认值常数，$C_{1\varepsilon}$=1.44，$C_{2\varepsilon}$=1.92，$C_{3\varepsilon}$ =0.09，湍动能 k 与耗散率 ε 的湍流普朗特数分别为 σ_k=1.0，σ_ε=1.3。

3) 重整化群 k-ε 模型

重整化群 k-ε 模型是对瞬时的 Navier-Stokes 方程用重整化群的数学方法推导出来的模型。模型中的常数与标准 k-ε 模型不同，而且方程中也出现了新的函数或者项。其湍动能与耗散率方程与标准 k-ε 模型有相似的形式：

$$\rho \frac{\mathrm{d}k}{\mathrm{d}t} = \frac{\partial}{\partial x_i}\left[(\alpha_k \mu_{\text{eff}})\frac{\partial k}{\partial x_i}\right] + G_k + G_b - \rho\varepsilon - Y_M \tag{1-63}$$

$$\rho \frac{\mathrm{d}\varepsilon}{\mathrm{d}t} = \frac{\partial}{\partial x_i}\left[(\alpha_\varepsilon \mu_{\text{eff}})\frac{\partial \varepsilon}{\partial x_i}\right] + C_{1\varepsilon}\frac{\varepsilon}{k}(G_k + C_{3\varepsilon}G_b) - C_{2\varepsilon}\rho\frac{\varepsilon^2}{k} - R \tag{1-64}$$

式中：G_k 表示由于平均速度梯度引起的湍动能产生，G_b 表示由于浮力影响引起的湍动能产生；Y_M 表示可压缩湍流脉动膨胀对总的耗散率的影响，这些参数与标准 k - ε 模型中相同。α_k 和 α_ε 分别是湍动能 k 和耗散率 ε 的有效湍流普朗特数的倒数。湍流粘性系数计算公式为 $\mathrm{d}\left(\frac{\rho^2 k}{\sqrt{\varepsilon\mu}}\right)=1.72\frac{\tilde{v}}{\sqrt{\tilde{v}^3-1-Cv}}\mathrm{d}\tilde{v}$，其中，$\tilde{v}=\mu_{\mathrm{eff}}/\mu$，$C_v\approx 100$。对于前面方程的积分，可以精确到有效雷诺数(涡旋尺度)对湍流输运的影响，这有助于处理低雷诺数和近壁流动问题的模拟。对于高雷诺数，上面方程可以给出：$\mu_t=\rho C_\mu\frac{k^2}{\varepsilon}$，$C_\mu=0.0845$。这个结果非常有意思，和标准 k - ε 模型的半经验推导给出的常数 $C_\mu=0.09$ 非常近似。在 Fluent 中，如果是默认设置，用重整化群 k - ε 模型时是针对的高雷诺数流动问题。如果对低雷诺数问题进行数值模拟，必须进行相应的设置。

4) 可实现 k - ε 模型

可实现 k - ε 模型的湍动能及其耗散率输运方程为

$$\rho\frac{\mathrm{d}k}{\mathrm{d}t}=\frac{\partial}{\partial x_i}\left[\left(\mu+\frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_i}\right]+G_k+G_b-\rho\varepsilon-Y_M \tag{1-65}$$

$$\rho\frac{\mathrm{d}\varepsilon}{\mathrm{d}t}=\frac{\partial}{\partial x_i}\left[\left(\mu+\frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_i}\right]+\rho C_1 S\varepsilon-\rho C_2\frac{\varepsilon^2}{k+\sqrt{v\varepsilon}}+C_{1\varepsilon}\frac{\varepsilon}{k}C_{3\varepsilon}G_b \tag{1-66}$$

式中：$C_1=\max\left[0.43,\frac{\eta}{\eta+5}\right]$，$\eta=Sk/\varepsilon$。

在上述方程中，G_k 表示由于平均速度梯度引起的湍动能产生，G_b 表示由于浮力影响引起的湍动能产生；Y_M 表示可压缩湍流脉动膨胀对总的耗散率的影响；C_2 和 $C_{1\varepsilon}$ 是常数；σ_k 和 σ_ε 分别是湍动能及其耗散率的湍流普朗特数。在 Fluent 中，作为默认值常数，$C_{1\varepsilon}$=1.44，C_2=1.9，σ_k=1.0，σ_ε=1.2。

该模型的湍流粘性系数与标准 k - ε 模型相同。不同的是，粘性系数中的 C_μ 不是常数，而是通过公式计算得到 $C_\mu=\frac{1}{A_0+A_s\frac{U^* K}{\varepsilon}}$，其中，$U^*=\sqrt{S_{ij}S_{ij}+\tilde{\Omega}_{ij}\tilde{\Omega}_{ij}}$，$\tilde{\Omega}_{ij}=\Omega_{ij}-2\varepsilon_{ijk}\omega_k$，$\Omega_{ij}=\bar{\Omega}_{ij}+2\varepsilon_{ijk}\omega_k$，$\tilde{\Omega}_{ij}$ 表示在角速度 ω_k 旋转参考系下的平均旋转张量率。模型常数 $A_0=4.04$，$A_s=\sqrt{6}\cos\phi$，$\phi=\frac{1}{3}\arccos(\sqrt{6}W)$，式中 $W=\frac{S_{ij}S_{jk}S_{ki}}{\tilde{S}}$，$\tilde{S}\equiv\sqrt{S_{ij}S_{ij}}$，$S_{ij}=\frac{1}{2}\left(\frac{\partial u_j}{\partial x_i}+\frac{\partial u_i}{\partial x_j}\right)$。从这些式子中发现，$C_\mu$ 是平均应变率与旋度的函数。在平衡边界层惯性底层，可以得到 $C_\mu=0.09$，与标准 k - ε 模型中采用的常数一样。

该模型适合的流动类型比较广泛，包括有旋均匀剪切流、自由流(射流和混合层)、腔道流动和边界层流动。对以上流动过程模拟结果都比标准 k - ε 模型的结果好，特别是可实现 k - ε 模型对圆口射流和平板射流模拟中，能给出较好的射流扩张角。

双方程模型中，无论是标准 k - ε 模型、重整化群 k - ε 模型还是可实现 k - ε 模型，三个模型有类似的形式，即都有 k 和 ε 的输运方程，它们的区别在于：①计算湍流粘性的方法

不同；②控制湍流扩散的湍流普朗特数不同；③ε 方程中的产生项和 G_k 关系不同。但都包含了相同的表示由于平均速度梯度引起的湍动能产生 G_k，表示由于浮力影响引起的湍动能产生 G_b；表示可压缩湍流脉动膨胀对总的耗散率的影响 Y_M。

湍动能产生项

$$G_k = -\rho \overline{u_i' u_j'} \frac{\partial u_j}{\partial x_i} \tag{1-67}$$

$$G_b = \beta g_i \frac{\mu_t}{P_{rt}} \frac{\partial T}{\partial x_i} \tag{1-68}$$

式中：P_{rt} 是能量的湍流普特朗数，对于可实现 k-ε 模型，默认设置值为 0.85；对于重整化群 k-ε 模型，$P_{rt} = 1/\alpha$，$\alpha = 1/P_{rt} = k/\mu C_p$。热膨胀系数 $\beta = -\frac{1}{\rho}\left(\frac{\partial \rho}{\partial T}\right)_p$，对于理想气体，浮力引起的湍动能产生项变为

$$G_b = -g_i \frac{\mu_t}{\rho P_{rt}} \frac{\partial \rho}{\partial x_i} \tag{1-69}$$

5) 雷诺应力模型

雷诺应力模型(RSM)是求解雷诺应力张量的各个分量的输运方程。具体形式为

$$\begin{aligned} \frac{\partial}{\partial t}(\rho \overline{u_i u_j}) + \frac{\partial}{\partial x_k}(\rho U_k \overline{u_i u_j}) = -\frac{\partial}{\partial x_k}[\rho \overline{u_i u_j u_k} + \overline{p(\delta_{kj} u_i + \delta_{ik} u_j)}] + \\ \frac{\partial}{\partial x_k}\left(\mu \frac{\partial}{\partial x_k} \overline{u_i u_j}\right) - \rho\left(\overline{u_i u_k} \frac{\partial U_j}{\partial x_k} + \overline{u_j u_k} \frac{\partial U_i}{\partial x_k}\right) - \rho\beta(g_i \overline{u_j \theta} + g_j \overline{u_i \theta}) + \\ p\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - 2\mu \overline{\frac{\partial u_i}{\partial x_k} \frac{\partial u_j}{\partial x_k}} - 2\rho \Omega_k (\overline{u_j u_m} \varepsilon_{ikm} + \overline{u_i u_m} \varepsilon_{jkm}) \end{aligned} \tag{1-70}$$

式中：左边的第二项是对流项 C_{ij}，右边第一项是湍流扩散项 D_{ij}^T，第二项是分子扩散项 D_{ij}^L，第三项是应力产生项 P_{ij}，第四项是浮力产生项 G_{ij}，第五项是压力应变项 ϕ_{ij}，第六项是耗散项 ε_{ij}，第七项系统旋转产生项 F_{ij}。

在式(1-69)中，C_{ij}、D_{ij}^L、P_{ij}、F_{ij} 不需要模拟，而 D_{ij}^T、G_{ij}、ϕ_{ij}、ε_{ij} 需要模拟以封闭方程。下面简单对几个需要模拟项进行模拟。

D_{ij}^T 可以用 Delay 和 Harlow 的梯度扩散模型来模拟，但这个模型会导致数值不稳定，在 Fluent 中是采用标量湍流扩散模型：

$$D_{ij}^T = \frac{\partial}{\partial x_k}\left(\frac{\mu_t}{\sigma_k} \frac{\partial \overline{u_i u_j}}{\partial x_k}\right) \tag{1-71}$$

式中：湍流粘性系数用 $\mu_t = \rho C_\mu \frac{k^2}{\varepsilon}$ 来计算，根据 Lien 和 Leschziner，$\sigma_k = 0.82$，这和标准 k-ε 模型中选取 1.0 有所不同。

压力应变项 ϕ_{ij} 可以分解为三项，即

$$\phi_{ij} = \phi_{ij,1} + \phi_{ij,2} + \phi_{ij}^{w} \tag{1-72}$$

式中：$\phi_{ij,1}$、$\phi_{ij,2}$ 和 ϕ_{ij}^{w} 分别是慢速项、快速项和壁面反射项，具体表述可以参见文献[2]。

浮力引起的产生项 G_{ij} 模拟为

$$G_{ij} = \beta \frac{\mu_t}{P_{rt}} \left(g_i \frac{\partial T}{\partial x_j} + g_j \frac{\partial T}{\partial x_i} \right) \tag{1-73}$$

耗散张量 ε_{ij} 模拟为

$$\varepsilon_{ij} = \frac{2}{3} \delta_{ij} (\rho\varepsilon + Y_M) \tag{1-74}$$

式中：$Y_M = 2\rho\varepsilon M_t^2$，$M_t$ 是马赫数；标量耗散率 ε 用标准 k-ε 模型中采用的耗散率输运方程求解。

6) 大涡模拟

湍流中包含了不同时间与长度尺度的涡旋。最大长度尺度通常为平均流动的特征长度尺度。最小尺度为 Komogrov 尺度。LES 的基本假设是：①动量、能量、质量及其他标量主要由大涡输运；②流动的几何和边界条件决定了大涡的特性，而流动特性主要在大涡中体现；③小尺度涡旋受几何和边界条件影响较小，并且各向同性，大涡模拟(LES)过程中，直接求解大涡，小尺度涡旋模拟，从而使得网格要求比 DNS 低。

LES 的控制方程是对 Navier-Stokes 方程在波数空间或者物理空间进行过滤得到的。过滤的过程是去掉比过滤宽度或者给定物理宽度小的涡旋，从而得到大涡旋的控制方程：

$$\frac{\partial \rho}{\partial t} + u \frac{\partial \rho \overline{u}_i}{\partial x_i} = 0 \tag{1-75}$$

$$\frac{\partial}{\partial t}(\rho \overline{u}_i) + \frac{\partial}{\partial x_j}(\rho \overline{u_i u_j}) = \frac{\partial}{\partial x_j}(\mu \frac{\partial \overline{u}_i}{\partial x_j}) - \frac{\partial \overline{p}}{\partial x_j} - \frac{\partial \tau_{ij}}{\partial x_j} \tag{1-76}$$

式中：τ_{ij} 为亚网格应力，$\tau_{ij} = \rho \overline{u_i u_j} - \rho \overline{u}_i \cdot \overline{u}_j$。

很明显，上述方程与雷诺平均方程很相似，只不过大涡模拟中的变量是过滤过的量，而非时间平均量，并且湍流应力也不同。

1.2.3 初始条件和边界条件

计算流体动力学(CFD)分析中，初始条件和边界条件的正确设置是关键的一步。现有的 CFD 软件都提供了现成的各种类型的边界条件，这里对有关的初始条件和边界条件作一般讨论。

1. 初始条件

顾名思义，初始条件就是计算初始给定的参数，即 $t = t_0$ 时给出各未知量的函数分布，如

$$\begin{cases} u = u(x,y,z,t_0) = u_0(x,y,z) \\ v = v(x,y,z,t_0) = v_0(x,y,z) \\ w = w(x,y,z,t_0) = w_0(x,y,z) \\ p = p(x,y,z,t_0) = p_0(x,y,z) \\ \rho = \rho(x,y,z,t_0) = \rho_0(x,y,z) \\ T = T(x,y,z,t_0) = T_0(x,y,z) \end{cases} \tag{1-77}$$

很明显，当流体运动定常时，无初始条件问题。

2. 边界条件

所谓边界条件就是流体力学方程组在求解域的边界上，流体物理量应满足的条件。例如，流体被固壁所限，流体将不应有穿过固壁的速度分量；在水面这个边界上，大气压强认为是常数(一般在距离不大的范围内可如此)；在流体与外界无热传导的边界上，流体与边界之间无温差，如此等。由于各种具体问题不同，边界条件提法千差万别，一般要保持恰当：①保持在物理上是正确的；②要在数学上不多不少，刚好能用来确定积分微分方程中的积分常数，而不是矛盾的或有随意性。

通常流体边界分为流固交界面和流流(液液、液气)交界面，下面分别讨论。

1) 流固分界面边界条件

飞机、船舶在空气及水中运动时的流固分界面，水在岸边及底部的流固分界面，均属这一类。一般而言，流体在固体边界上的速度依流体有无粘性而定。对于粘性流体，流体将粘附于固体表面(无滑移)，即

$$v|_F = v|_S \tag{1-78}$$

式中：$v|_F$ 是流体速度；$v|_S$ 是固壁面相应点的速度。式(1-78)表明，在流固边界面上，流体在一点的速度等于固体在该点的速度。对于无粘性流体，流体可沿界面滑移，即有速度的切向分量，但不能离开界面，也就是流体的法向速度分量等于固体的法向速度分量，即

$$v_n|_F = v|_S \tag{1-79}$$

另外，也可视所给条件，给出无温差条件：

$$T|_F = T|_S \tag{1-80}$$

式中：$T|_F$ 是流体温度，$T|_S$ 是固壁面相应点的温度。

2) 液液分界面边界条件

密度不同的两种液体的分界面就属于这一类。一般而言，对分界面两侧的液体情况经常给出的条件是

$$v_1 = v_2, T_1 = T_2, p_1 = p_2 \tag{1-81}$$

对应力及传导热情况给出的条件是

$$\tau = \mu_1 \frac{\partial u}{\partial n}|_1 = \mu_2 \frac{\partial u}{\partial n}|_2 \tag{1-82}$$

$$Q = k_1 \frac{\partial T}{\partial n}|_1 = k_2 \frac{\partial T}{\partial n}|_2 \tag{1-83}$$

3) 液气分界面边界条件

液气分界面最典型的是水与大气的分界面，即自由面。由于自由面本身是运动和变形的，而且其形状常常也是一个需要求解的未知函数，因此就有一个自由面的运动学条件问题。设自由面方程为

$$F(x,y,z,t) = 0 \tag{1-84}$$

并假定在自由面上的流体质点始终保持在自由面上，则流体质点在自由面上一点的法向速度，应该等于自由面本身在这一点的法向速度。经过一系列推导(参见文献[2])，得到自由液面运动学条件：

$$\frac{\partial F}{\partial t} + v \cdot \nabla F = 0 \tag{1-85}$$

如果要考虑液气边界上的表面张力，则在界面两侧，两种介质的压强差与表面张力有如下关系：

$$p_2 - p_1 = \sigma\left(\frac{1}{R_1} + \frac{1}{R_2}\right) \tag{1-86}$$

这就是自由面上的动力学条件。当不考虑表面张力时，有

$$p = p_a \tag{1-87}$$

式中：p_a 为大气压强。

4)　无限远的条件

流体力学中的很多问题，流体域是无限远的。例如，飞机在空中飞行时，流体是无界的。如果将坐标系取在运动物体上，这时无限远处的边界条件为

当 $x \to \infty$ 时，

$$u = u_\infty,\ p = p_\infty \tag{1-88}$$

其中下标 ∞ 表示无穷远处的值。

1.3 CFD 模型的离散——有限体积法

1.3.1 CFD 模型的数值求解方法概述

从上面的分析看到，CFD 模型(控制方程)是一系列偏微分方程组，要得到解析解比较困难，目前，均采用数值方法得到其满足实际需要的近似解。

数值方法求解 CFD 模型的基本思想是：把原来在空间与时间坐标中连续的物理量的场(如速度场、温度场、浓度场等)，用一系列有限个离散点(称为节点，node)上的值的集合来代替，通过一定的原则建立起这些离散点上变量值之间关系的代数方程(称为离散方程，discretization equation)，求解所建立起来的代数方程以获得所求解变量的近似解。在过去的几十年内已经发展了多种数值解法，其间的主要区别在于区域的离散方式、方程的离散方式及代数方程求解的方法这三个环节上。在 CFD 求解计算中用得较多的数值方法有：有限差分法(finite difference method，FDM)、有限体积法(finite volume method，FVM)、有限元法(finite element method，FEM)及有限分析法(finite analytic method，FAM)。下面简要介绍，后面将着重介绍有限体积法。

1. 有限差分法

有限差分法是历史上采用最早的数值方法，对简单几何形状中的流动与换热问题也是一种最容易实施的数值方法。其基本点是：将求解区域用与坐标轴平行的一系列网格线的交点所组成的点的集合来代替，在每个节点上，将控制方程中每一个导数用相应的差分表达式来代替，从而在每个节点上形成一个代数方程，每个方程中包括了本节点及其附近一些节点上的未知值，求解这些代数方程就获得了所需的数值解。由于各阶导数的差分表达式可以从 Taylor(泰勒)展开式来导出，这种方法又称建立离散方程的 Taylor 展开法。

有限差分法软件一般研究者自己编写，很少看到商品的有限差分法软件。

2. 有限体积法

在有限体积法中将所计算的区域划分成一系列控制体积，每个控制体积都有一个节点作代表，通过将守恒型的控制方程对控制体积作积分来导出离散方程。在导出过程中，需要对界面上的被求函数本身及其一阶导数的构成作出假定，这种构成的方式就是有限体积法中的离散格式。用有限体积法导出的离散方程可以保证具有守恒特性，而且离散方程系数的物理意义明确，是目前流动与传热问题的数值计算中应用最广泛的一种方法。

Phoenics 是最早投入市场的有限体积法软件，Fluent、STAR-CD 和 CFX 都是常用的有限体积法软件，它们在流动、传热传质、燃烧和辐射等方面应用广泛。

3. 有限元法

在有限元法中把计算区域划分成一系列单元体(在二维情况下，单元体多为三角形或四边形)，在每个单元体上取数个点作为节点，然后通过对控制方程做积分来获得离散方程。它与有限体积法区别主要在于如下两点。

(1) 要选定一个形状函数(最简单的是线性函数)，并通过单元体中节点上的被求变量之值来表示该形状函数，在积分之前将该形状函数代入到控制方程中去。这一形状函数在建立离散方程及求解后结果的处理上都要应用。

(2) 控制方程在积分之前要乘上一个权函数，要求在整个计算区域上控制方程余量(即代入形状函数后使控制方程等号两端不相等的差值)的加权平均值等于零，从而得出一组关于节点上的被求变量的代数方程组。

有限元法的最大优点是对不规则区域的适应性好。但计算的工作量一般较有限体积法大，而且在求解流动与换热问题时，对流项的离散处理方法及不可压流体原始变量法求解方面没有有限体积法成熟。

Ansys、Sysweld 和北京飞箭公司的 FEPG(finite element programs generator)等有限元软件比较流行。

4. 有限分析法

有限分析法是由美籍华裔科学家陈景仁教授在 1981 年提出的。在这种方法中，也像有限差分法那样，用一系列网格线将区域离散，所不同的是每个节点与相邻的 4 个网格(二维)问题组成计算单元，即一个计算单元由一个中心节点与 8 个邻点组成。在计算单元中把控制方程中的非线性项(如 Navier-Stokes 方程中的对流项)局部线性化(即认为流速已知)，并对该单元上未知函数的变化型线作出假设，把所选定型线表达式中的系数和常数项用单元边界节点上未知的变量值来表示，这样该单元内的被求问题就转化为第一类边界条件下的一个定解问题，可以找出其分析解；然后利用这一分析解，得出该单元中点及边界上 8 个邻点上未知值间的代数方程，此即为单元中点的离散方程。有限分析法中的系数不像有限体积法中那样有明确的物理意义，对不规则区域的适应性也较差。

1.3.2 有限体积法

从上面的简介看到，有限体积法是一种分块近似的计算方法，其中比较重要步骤是计算区域的离散和控制方程的离散。

1. 计算区域的离散化

所谓区域的离散化(domain discretization)实质上就是用一组有限个离散的点来代替原来的连续空间。一般的实施过程是：把所计算的区域划分成许多个互不重叠的子区域(sub-domain)，确定每个子区域中的节点位置及该节点所代表的控制体积。区域离散后，得到以下四种几何要素。

- 节点(node)：需要求解的未知物理量的几何位置。
- 控制体积(control volume)：应用控制方程或守恒定律的最小几何单位。
- 界面(face)：它定义了与各节点相对应的控制体积的界面位置。
- 网格线(grid line)：连接相邻两节点面形成的曲线簇。

一般把节点看成是控制体积的代表。在离散过程中，将一个控制体积上的物理量定义并存储在该节点处。图 1-2 给出了一维问题的有限体积法计算网格，图 1-3 给出了二维问题的有限体积法计算网格。

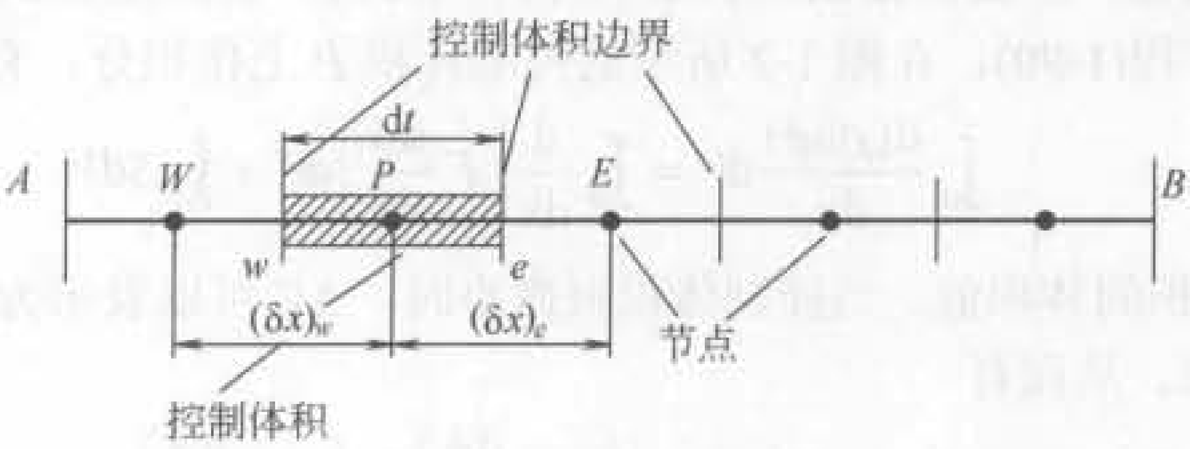


图 1-2　一维的有限体积法网格

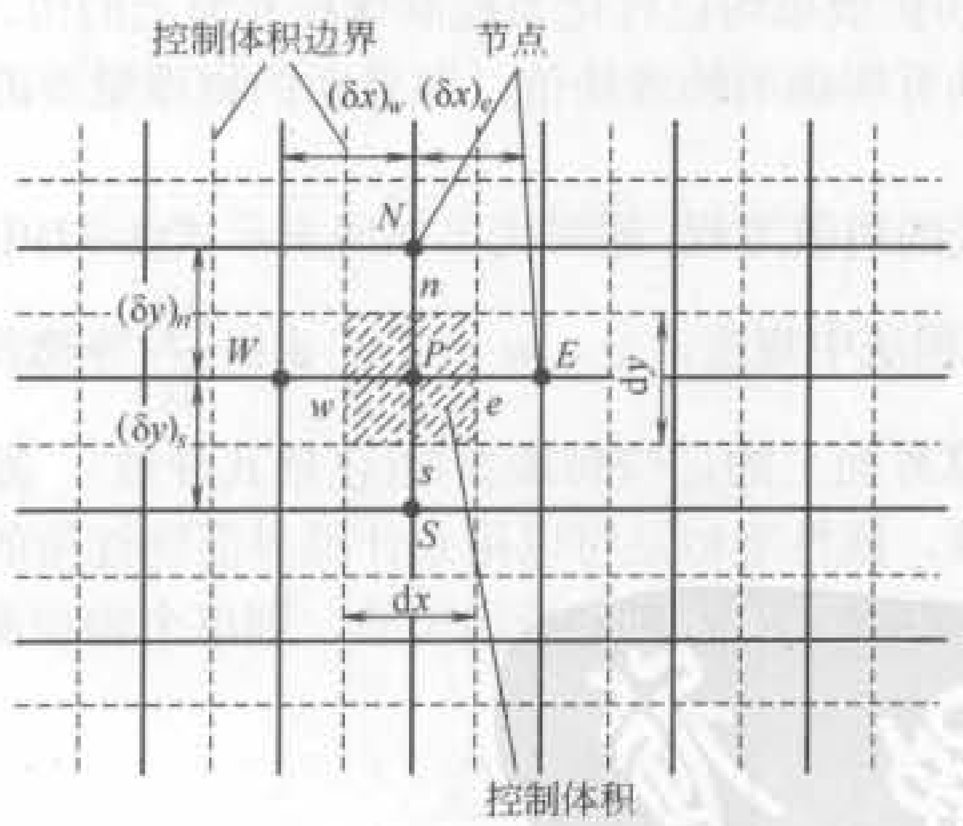


图 1-3　二维的有限体积法网格

计算区域离散的网格有两类：结构化网格和非结构化网格。节点排列有序，即当给出了一个节点的编号后，立即可以得出其相邻节点的编号，所有内部节点周围的网格数目相同。这种网格称为结构化网格(structured grid)。结构化网格具有实现容易、生成速度快、网格质量好、数据结构简单的优点，但不能实现复杂边界区域的离散。

而非结构化网格的内部节点以一种不规则的方式布置在流场中，各节点周围的网格数目不尽相同。这种网格虽然生成过程比较复杂，但却有极大的适应性，对复杂边界的流场

计算问题特别有效。

2. 控制方程的离散化

前面给出的流体流动问题的控制方程，无论是连续性方程、动量方程，还是能量方程，都可写成如式(1-89)所示的通用形式，

$$\frac{\partial(\rho\phi)}{\partial t}+\mathrm{div}(\rho u\phi)=\mathrm{div}(\Gamma\,\mathrm{grad}\phi)+S \tag{1-89}$$

对于一维稳态问题，其控制方程如式(1-90)所示：

$$\frac{\mathrm{d}(\rho u\phi)}{\mathrm{d}x}=\frac{\mathrm{d}}{\mathrm{d}x}\left(\Gamma\frac{\mathrm{d}\phi}{\mathrm{d}x}\right)+S \tag{1-90}$$

式中：从左到右各项分别为对流项、扩散项和源项。方程中的ϕ是广义变量，可以为速度、温度或浓度等一些待求的物理量。Γ是相应于ϕ的广义扩散系数，S是广义源项。变量ϕ在端点A和B的边界值为已知。

有限体积法的关键一步是在控制体积上积分控制方程，在控制体积节点上产生离散的方程。对一维模型方程(1-90)，在图 1-2 所示的控制体积P上作积分，有

$$\int_{\Delta V}\frac{\mathrm{d}(\rho u\phi)}{\mathrm{d}x}\mathrm{d}V=\int_{\Delta V}\frac{\mathrm{d}}{\mathrm{d}x}\left(\Gamma\frac{\mathrm{d}\phi}{\mathrm{d}x}\right)\mathrm{d}V+\int_{\Delta V}S\mathrm{d}V \tag{1-91}$$

式中：ΔV是控制体积的体积值。当控制体积很微小时，ΔV可以表示为$\Delta V\cdot A$，这里A是控制体积界面的面积。从而有

$$(\rho u\phi A)_e-(\rho u\phi A)_w=\left(\Gamma A\frac{\mathrm{d}\phi}{\mathrm{d}x}\right)_e-\left(\Gamma A\frac{\mathrm{d}\phi}{\mathrm{d}x}\right)_w+S\cdot\Delta V \tag{1-92}$$

从式(1-92)看到，对流项和扩散项均已转化为控制体积界面上的值。有限体积法最显著的特点之一就是离散方程中具有明确的物理插值，即界面的物理量要通过插值的方式由节点的物理量来表示。

为了建立所需要形式的离散方程，需要找出如何表示式(1-92)中界面e和w处的ρ、u、Γ、ϕ和$\frac{\mathrm{d}\phi}{\mathrm{d}x}$。在有限体积法中规定，$\rho$、$u$、$\Gamma$、$\phi$和$\frac{\mathrm{d}\phi}{\mathrm{d}x}$等物理量均是在节点处定义和计算的。因此，为了计算界面上的这些物理参数(包括其导数)，需要一个物理参数在节点间的近似分布。可以想象，线性近似是可以用来计算界面物性值的最直接，也是最简单的方式。这种分布叫做中心差分。如果网格是均匀的，则单个物理参数(以扩散系数Γ为例)的线性插值结果是

$$\begin{cases}\Gamma_e=\dfrac{\Gamma_P+\Gamma_E}{2}\\[2mm]\Gamma_w=\dfrac{\Gamma_W+\Gamma_P}{2}\end{cases} \tag{1-93}$$

$(\rho u\phi A)$的线性插值结果是

$$\begin{cases}(\rho u\phi A)_e=(\rho u)_e A_e\dfrac{\phi_P+\phi_E}{2}\\[2mm](\rho u\phi A)_w=(\rho u)_w A_w\dfrac{\phi_W+\phi_P}{2}\end{cases} \tag{1-94}$$

与梯度项相关的扩散通量的线性插值结果是

$$\begin{cases}\left(\Gamma A\dfrac{\mathrm{d}\phi}{\mathrm{d}x}\right)_e=\Gamma_e A_e\left[\dfrac{\phi_E-\phi_P}{(\delta x)_e}\right]\\ \left(\Gamma A\dfrac{\mathrm{d}\phi}{\mathrm{d}x}\right)_w=\Gamma_w A_w\left[\dfrac{\phi_P-\phi_W}{(\delta x)_w}\right]\end{cases}\tag{1-95}$$

对于源项 S，它通常是时间和物理量 ϕ 的函数。为了简化处理，将 S 转化为如下线性方式：

$$S=S_C+S_P\phi_P\tag{1-96}$$

式中：S_C 是常数，S_P 是随时间和物理量 ϕ 变化的项。将式(1-93)～式(1-96)代入方程(1-92)，有

$$\begin{aligned}&(\rho u)_e A_e\frac{\phi_P+\phi_E}{2}-(\rho u)_w A_w\frac{\phi_W+\phi_P}{2}\\&=\Gamma_e A_e\left[\frac{\phi_E-\phi_P}{(\delta x)_e}\right]-\Gamma_w A_w\left[\frac{\phi_P-\phi_W}{(\delta x)_w}\right]+(S_C+S_P\phi_P)\cdot\Delta V\end{aligned}\tag{1-97}$$

整理后得

$$\begin{aligned}&\left[\frac{\Gamma_e}{(\delta x)_e}A_e+\frac{\Gamma_w}{(\delta x)_w}A_w-S_P\cdot\Delta V\right]\phi_P\\&=\left[\frac{\Gamma_w}{(\delta x)_w}A_w+\frac{(\rho u)_w}{2}A_w\right]\phi_W+\left[\frac{\Gamma_e}{(\delta x)_e}A_e-\frac{(\rho u)_e}{2}A_e\right]\phi_E+S_C\cdot\Delta V\end{aligned}$$

记为

$$a_P\phi_P=a_W\phi_W+a_E\phi_E+b\tag{1-98}$$

式中：

$$\begin{cases}a_W=\dfrac{\Gamma_w}{(\delta x)_w}A_w+\dfrac{(\rho u)_w}{2}A_w\\ a_E=\dfrac{\Gamma_e}{(\delta x)_e}A_e-\dfrac{(\rho u)_e}{2}A_e\\ a_P=\dfrac{\Gamma_e}{(\delta x)_e}A_e+\dfrac{\Gamma_w}{(\delta x)_w}A_w-S_P\cdot\Delta V\\ \quad=a_E+a_W+\dfrac{(\rho u)_e}{2}A_e-\dfrac{(\rho u)_w}{2}A_w-S_P\cdot\Delta V\\ b=S_C\cdot\Delta V\end{cases}\tag{1-99}$$

对于一维问题，控制体积界面 e 和 w 处的面积 A_e 和 A_w 均为 1，即单位面积。这样 $\Delta V=\Delta x$，式(1-99)中各系数可转化为

$$\begin{cases} a_W = \dfrac{\Gamma_w}{(\delta x)_w} + \dfrac{(\rho u)_w}{2} \\ a_E = \dfrac{\Gamma_e}{(\delta x)_e} - \dfrac{(\rho u)_e}{2} \\ a_P = a_E + a_W + \dfrac{(\rho u)_e}{2} - \dfrac{(\rho u)_w}{2} - S_P \cdot \Delta x \\ b = S_C \cdot \Delta x \end{cases} \tag{1-100}$$

方程(1-98)即为方程(1-90)的离散形式，每个节点上都可建立此离散方程，通过求解方程组，就可得到各物理量在各节点处的值。

为了后续讨论的方便，定义两个新的物理量 F 和 D，其中 F 表示通过界面上单位面积的对流质量通量(convective mass flux)，简称对流质量流量，D 表示界面的扩散传导性(diffusion conductance)。定义表达式如下：

$$\begin{cases} F = \rho u \\ D = \dfrac{\Gamma}{\delta x} \end{cases} \tag{1-101}$$

这样，F 和 D 在控制界面上的值分别为

$$\begin{cases} F_w = (\rho u)_w, F_e = (\rho u)_e \\ D_w = \dfrac{\Gamma_w}{(\delta x)_w}, D_e = \dfrac{\Gamma_e}{(\delta x)_e} \end{cases} \tag{1-102}$$

在此基础上，定义一维单元的 Peclet 数 Pe 如下：

$$Pe = \frac{F}{D} = \frac{\rho u}{\Gamma / \delta x} \tag{1-103}$$

式中：Pe 表示对流与扩散的强度之比。当 Pe 数为 0 时，对流-扩散演变为纯扩散问题，即流场中没有流动，只有扩散；当 $Pe>0$ 时，流体沿 x 方向流动，当 Pe 数很大时，对流-扩散问题演变为纯对流问题。一般在中心差分格式中，有 $Pe<2$ 的要求。

将式(1-101)代入方程(1-100)，有

$$\begin{cases} a_W = D_w + \dfrac{F_w}{2} \\ a_E = D_e - \dfrac{F_e}{2} \\ a_P = a_E + a_W + \dfrac{F_e}{2} - \dfrac{F_w}{2} - S_P \cdot \Delta x \\ b = S_C \cdot \Delta x \end{cases} \tag{1-104}$$

对于瞬态问题，与稳态问题相似，主要是瞬态项的离散。其一维瞬态问题的通用控制方程如下：

$$\frac{\partial(\rho\phi)}{\partial t} + \frac{\partial(\rho u \phi)}{\partial x} = \frac{\partial}{\partial x}\left(\Gamma \frac{\partial \phi}{\partial x}\right) + S \tag{1-105}$$

该方程是一个包含瞬态及源项的对流-扩散方程。从左到右，方程中的各项分别是瞬态项、对流项、扩散项及源项。方程中的 ϕ 是广义变量，如速度分量、温度、浓度等，Γ 为相应于 ϕ 的广义扩散系数，S 为广义源项。

对于瞬态问题用有限体积法求解时，在将控制方程对控制体积作空间积分的同时，还必须对时间间隔 Δt 作时间积分。对控制体积所作的空间积分与稳态问题相同，这里仅叙述对时间的积分。

将方程(1-105)在一维计算网格上对时间及控制体积进行积分，有

$$\begin{aligned}&\int_t^{t+\Delta t}\int_{\Delta V}\frac{\partial(\rho\phi)}{\partial t}\mathrm{d}V\mathrm{d}t+\int_t^{t+\Delta t}\int_{\Delta V}\frac{\partial(\rho u\phi)}{\partial x}\mathrm{d}V\mathrm{d}t\\&=\int_t^{t+\Delta t}\int_{\Delta V}\frac{\partial}{\partial x}\left(\Gamma\frac{\partial\phi}{\partial x}\right)\mathrm{d}V\mathrm{d}t+\int_t^{t+\Delta t}\int_{\Delta V}S\mathrm{d}V\mathrm{d}t\end{aligned}\tag{1-106}$$

改写后，有

$$\begin{aligned}&\int_{\Delta V}\left[\int_t^{t+\Delta t}\frac{\partial(\rho\phi)}{\partial t}\mathrm{d}t\right]\mathrm{d}V+\int_t^{t+\Delta t}[(\rho u\phi A)_e-(\rho u\phi A)_w]\mathrm{d}t\\&=\int_t^{t+\Delta t}\left[\left(\Gamma A\frac{\mathrm{d}\phi}{\mathrm{d}x}\right)_e-\left(\Gamma A\frac{\mathrm{d}\phi}{\mathrm{d}x}\right)_w\right]\mathrm{d}t+\int_t^{t+\Delta t}S\Delta V\mathrm{d}t\end{aligned}\tag{1-107}$$

式中：A 是图 1-2 中控制体积 P 的界面处的面积。

在处理瞬态项时，假定物理量 ϕ 在整个控制体积 P 上均具有节点处值 ϕ_P，并用线性插值 $(\phi_P-\phi_P^0)/\Delta t$ 来表示 $\partial\phi/\partial t$。源项也分解为线性方式 $S=S_c+S_P\phi_P$。对流项和扩散项的值按中心差分格式通过节点处的值来表示，则有

$$\begin{aligned}&\rho(\phi_P-\phi_P^0)\Delta V+\int_t^{t+\Delta t}\left[(\rho u)_e A_e\frac{\phi_P+\phi_E}{2}-(\rho u)_w A_w\frac{\phi_W+\phi_P}{2}\right]\mathrm{d}t\\&=\int_t^{t+\Delta t}\left\{\Gamma_e A_e\left[\frac{\phi_E-\phi_P}{(\delta x)_e}\right]-\Gamma_w A_w\left[\frac{\phi_P-\phi_W}{(\delta x)_W}\right]\right\}\mathrm{d}t+\int_t^{t+\Delta t}(S_c+S_P\phi_P)\Delta V\mathrm{d}t\end{aligned}\tag{1-108}$$

假定变量 ϕ_P 对时间的积分为

$$\int_t^{t+\Delta t}\phi_P\mathrm{d}t=[f\phi_P-(1-f)\phi_P{}^0]\Delta t\tag{1-109}$$

式中：上标 0 代表 t 时刻；ϕ_P 是时刻的值；f 为 0 与 1 之间的加权因子，当 f=0 时，变量取老值进行时间积分，当 f=1 时，变量采用新值进行时间积分。

将 ϕ_P、ϕ_E、ϕ_W 及 $S_C+S_P\phi_P$ 采用类似式(1-109)进行时间积分，式(1-108)可写成

$$\begin{aligned}&\rho(\phi_P-\phi_P^0)\frac{\Delta V}{\Delta t}+f\left[(\rho u)_e A_e\frac{\phi_P+\phi_E}{2}-(\rho u)_w A_w\frac{\phi_W+\phi_P}{2}\right]+\\&(1-f)\left[(\rho u)_e A_e\frac{\phi_P^0+\phi_E^0}{2}-(\rho u)_w A_w\frac{\phi_W^0+\phi_P^0}{2}\right]\\&=f\left\{\Gamma_e A_e\left[\frac{\phi_E-\phi_P}{(\delta x)_e}\right]-\Gamma_w A_w\left[\frac{\phi_P-\phi_W}{(\delta x)_W}\right]\right\}+\\&(1-f)\left\{\Gamma_e A_e\left[\frac{\phi_E^0-\phi_P^0}{(\delta x)_e}\right]-\Gamma_w A_w\left[\frac{\phi_P^0-\phi_W^0}{(\delta x)_W}\right]\right\}+\\&[f(S_C+S_P\phi_P)+(1-f)\times(S_c+S_P\phi_P^0)]\Delta V\end{aligned}\tag{1-110}$$

整理后得

$$\left\{\rho\frac{\Delta V}{\Delta t}+f\left[\frac{(\rho u)_e A_e}{2}-\frac{(\rho u)_w A_w}{2}\right]+f\left[\frac{\Gamma_e A_e}{(\delta x)_e}+\frac{\Gamma_w A_w}{(\delta x)_W}\right]-fS_P\Delta V\right\}\phi_P$$
$$=\left[\frac{(\rho u)_w A_w}{2}+\frac{\Gamma_w A_w}{(\delta x)_W}\right][f\phi_W+(1-f)\phi_W^0]+$$
$$\left[\frac{\Gamma_e A_e}{(\delta x)_e}-\frac{(\rho u)_e A_e}{2}\right][f\phi_E+(1-f)\phi_E^0]+ \tag{1-111}$$
$$\left\{\rho\frac{\Delta V}{\Delta t}-(1-f)\left[\frac{\Gamma_e A_e}{(\delta x)_e}+\frac{(\rho u)_e A_e}{2}\right]-\right.$$
$$\left.(1-f)\left[\frac{(\rho u)_w A_w}{2}-\frac{(\rho u)_w A_w}{2}\right]+(1-f)S_P\Delta V\right\}\phi_P^0+S_C\Delta V$$

同样引入稳态中关于符号 F 和 D 的定义，并将原来的定义作一定扩展，即乘以面积 A，有

$$\begin{cases} F_w=(\rho u)_w A_w, F_e=(\rho u)_e A_e \\ D_w=\dfrac{\Gamma_w A_w}{(\delta x)_w}, D_e=\dfrac{\Gamma_e A_e}{(\delta x)_e} \end{cases} \tag{1-112}$$

将式(1-112)代入方程(1-111)，得

$$\left[\rho\frac{\Delta V}{\Delta t}+f(D_e-D_w)+f\left(\frac{F_e}{2}+\frac{F_w}{2}\right)-fS_P\Delta V\right]\phi_P$$
$$=\left(\frac{F_w}{2}+D_w\right)[f\phi_W+(1-f)\phi_W^0]+\left(D_e-\frac{F_e}{2}\right)[f\phi_E+(1-f)\phi_E^0]+ \tag{1-113}$$
$$\left\{\rho\frac{\Delta V}{\Delta t}-(1-f)\left[D_e+\frac{F_e}{2}\right]-(1-f)\left[D_w-\frac{F_w}{2}\right]+(1-f)S_P\Delta V\right\}\phi_P^0+$$
$$S_C\Delta V$$

同样也像稳态问题引入 a_P、a_W 和 a_E，式(1-113)变为

$$a_P\phi_P=a_W[f\phi_W+(1-f)\phi_W^0]+a_E[f\phi_E+(1-f)\phi_E^0]+$$
$$\left\{\rho\frac{\Delta V}{\Delta t}-(1-f)\left[D_e+\frac{F_e}{2}\right]-(1-f)\left[D_w-\frac{F_w}{2}\right]+(1-f)S_P\Delta V\right\}\phi_P^0+ \tag{1-114}$$
$$S_C\Delta V$$

式中：

$$\begin{cases} a_P=a_P{}^0+f(a_E+a_W)+f(F_e-F_w)-fS_P\Delta V \\ a_W=D_w+\dfrac{F_w}{2} \\ a_E=D_e-\dfrac{F_e}{2} \\ a_P{}^0=\rho\dfrac{\Delta V}{\Delta t} \end{cases} \tag{1-115}$$

根据 f 的取值，瞬态问题对时间的积分有几种方案，当 f=0 时，变量的初值出现在方程(1-114)的右端，从而可直接求出在现时刻的未知变量值，这种方案称为显式时间积分方案。当 $0<f<1$ 时，现时刻的未知变量出现在方程(1-114)的两端，需要解若干个方程组成的方程组才能求出现时刻的变量值，这种方案称为隐式时间积分方案。当 f=1 时，就成为了全隐式时间积分方案了。当 f=1/2 时，称为 Crank-Nicolson 时间积分方案。

进一步将一维问题扩展为二维与三维问题。在二维问题中，计算区域离散如图 1-4 所示。发现只是增加第二坐标 y，控制体积增加的上下界面，分别用 n(north)和 s(south)表示，相应的两个邻点记为 N 和 S。在全隐式时间积分方案下的二维瞬态对流-扩散问题的离散方程为

$$a_P\phi_P = a_W\phi_W + a_E\phi_E + a_N\phi_N + a_S\phi_S + b \tag{1-116}$$

式中：

$$\begin{cases} a_P = a_P^0 + a_E + a_W + a_S + a_N + (F_e - F_w) + (F_n - F_s) - S_P\Delta V \\ a_W = D_w + \max(0, F_w) \\ a_E = D_e + \max(0, -F_e) \\ a_S = D_s + \max(0, F_s) \\ a_N = D_n + \max(0, -F_n) \\ a_P^0 = \rho_P^0 \dfrac{\Delta V}{\Delta t} \\ b = S_C\Delta V + a_P^0 + \phi_P^0 \end{cases} \tag{1-117}$$

在三维问题中，计算区域离散如图 1-4 所示(两个方向的投影)。在二维的基础上增加第三坐标 z，控制体积增加的前后界面，分别用 t(top)和 b(bottom)表示，相应的两个邻点记为 T 和 B。在全隐式时间积分方案下的三维瞬态对流-扩散问题的离散方程为

$$a_P\phi_P = a_W\phi_W + a_E\phi_E + a_N\phi_N + a_S\phi_S + a_T\phi_T + a_B\phi_B + b \tag{1-118}$$

式中：

$$\begin{cases} a_P = a_P{}^0 + a_E + a_W + a_S + a_N + (F_e - F_w) + (F_n - F_s) + \\ \qquad (F_t - F_b) - S_P\Delta V \\ a_W = D_w + \max(0, F_w) \\ a_E = D_e + \max(0, -F_e) \\ a_S = D_s + \max(0, F_s) \\ a_N = D_n + \max(0, -F_n) \\ a_T = D_t + \max(0, -F_t) \\ a_B = D_b + \max(0, F_b) \\ a_P^0 = \rho_P^0 \dfrac{\Delta V}{\Delta t} \\ b = S_C\Delta V + a_P^0 + \phi_P^0 \end{cases} \tag{1-119}$$

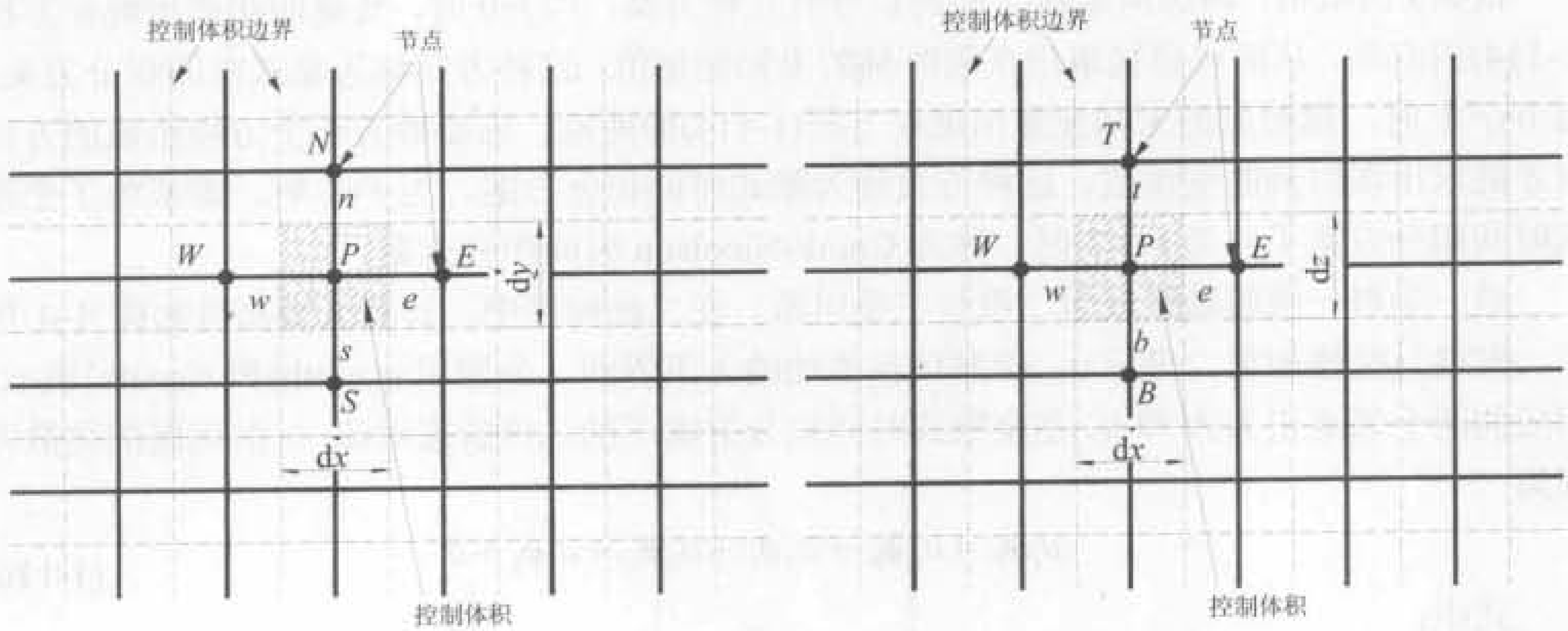


图 1-4

1.3.3 有限体积法中常用的离散格式

1. 常用的离散格式

有限体积法常用的离散格式有：中心差分格式、一阶迎风格式、混合格式、指数格式、乘方格式、二阶迎风格式、QUICK 格式。各种离散格式对一维、稳态、无源项的对流-扩散问题的通用控制方程(1-120)均能得到式(1-121)的形式。对于高阶情况如式(1-122)所示。

$$\frac{\mathrm{d}(\rho u\phi)}{\mathrm{d}x}=\frac{\mathrm{d}}{\mathrm{d}x}\left(\Gamma\frac{\mathrm{d}\phi}{\mathrm{d}x}\right) \tag{1-120}$$

$$a_P\phi_P = a_W\phi_W + a_E\phi_E \tag{1-121}$$

$$a_P\phi_P = a_W\phi_W + a_{WW}\phi_{WW} + a_E\phi_E + a_{EE}\phi_{EE} \tag{1-122}$$

式中，对于一阶情况，$a_P = a_W + a_E + (F_e - F_w)$，对于二阶情况，$a_P = a_W + a_E + a_{WW} + a_{EE} + (F_e - F_w)$，其中系数 a_W 和 a_E 取决于所使用的离散格式(高阶还有 a_{WW} 和 a_{EE})。为了便于比较和编程计算，将各种离散格式的系数总结于表 1-2 中。

表 1-2 不同离散格式下系数 a_W 和 a_E 的计算公式

离散格式	系数 a_W	系数 a_E
中心差分格式	$D_w + \frac{F_w}{2}$	$D_e - \frac{F_e}{2}$
一阶迎风格式	$D_w + \max(F_w, 0)$	$D_e + \max(0, -F_e)$
混合格式	$\max\left[F_w, \left(D_w + \frac{F_w}{2}\right), 0\right]$	$\max\left[-F_e, \left(D_e - \frac{F_e}{2}\right), 0\right]$
指数格式	$\frac{F_w \exp(F_w / D_w)}{\exp(F_w / D_w) - 1}$	$\frac{F_e}{\exp(F_e / D_e) - 1}$

续表

离散格式	系数 a_W	系数 a_E
乘方格式	$D_w \max[0,(1-0.1\|P_w\|)^5]+$ $\max[F_w,0]$	$D_e \max[0,(1-0.1\|P_e\|)^5]+$ $\max[-F_e,0]$
二阶迎风格式	$D_w+\frac{3}{2}\alpha F_w+\frac{1}{2}\alpha F_e$ $a_{WW}=-\frac{1}{2}\alpha F_w$	$D_e-\frac{3}{2}(1-\alpha)F_e-\frac{1}{2}(1-\alpha)F_w$ $a_{EE}=\frac{1}{2}(1-\alpha)F_e$
QUICK格式	$D_w+\frac{6}{8}\alpha_w F_w+\frac{1}{8}\alpha_w F_e+\frac{3}{8}(1-\alpha_w)F_w$ $a_{WW}=-\frac{1}{8}\alpha_w F_w$	$D_e-\frac{3}{8}\alpha_e F_e-\frac{6}{8}(1-\alpha_e)F_e-\frac{1}{8}(1-\alpha_e)F_w$ $a_{EE}=\frac{1}{8}(1-\alpha_e)F_e$

2. 建立离散方程应遵循的原则

在利用有限体积法建立离散方程时，必须遵守以下四条基本原则。

1)　控制体积界面上的连续性原则

当一个面为相邻的两个控制体积所共有时，在这两个控制体积的离散方程中，通过该界面的通量(包括热通量、质量通量、动量通量)的表达式必须相同。即通过某特定界面从一个控制体积所流出的热通量，必须等于通过该界面进入相邻控制体积的热通量，否则，总体平衡就得不到满足。

2)　正系数原则

中心节点系数 a_P 和相邻节点系数 a_{nb} 必须恒为正值。该原则是求得合理解的重要保证。当违背这一原则，结果往往是得到物理上不真实的解。例如，如果相邻节点的系数为负值，就可能出现边界温度的增加引起的相邻节点温度降低。

3)　源项的负斜率线性化原则

源项斜率为负可以保证正系数原则。从式(1-100)中看到，当相邻节点的系数皆为正值，但有源项 S_p 的存在，中心节点系数 a_P 仍有可能为负。当我们规定 $S_p \leqslant 0$，便可以保证 a_P 为正值。

4)　系数 a_P 等于相邻节点系数之和原则

当源项为0时，我们发现中心节点系数等于相邻节点系数之和，而当有源项存在时也应该保证这一原则，如果不能满足这个条件，可以取 S_P 为0。

1.4　流场数值计算算法分析

上面建立了控制方程的离散方程，即建立了可以数值计算的代数方程组。常规解法只能应付已知速度场求温度场分布这类简单的问题，所以对所生成的离散方程进行某种调整，

并且对各未知量(速度、压力、温度等)的求解顺序及方式进行特殊处理。为此，流场数值计算是针对常规解法存在的主要问题进行改善形成的一系列方法集。流场计算的基本过程是在空间上用有限体积法(或其他类似方法)将计算区域离散成许多小的体积单元，在每个体积单元上对离散后的控制方程组进行求解。其本质是对离散方程进行求解，一般可以分为分离解法(segregated method)和耦合解法(coupled method)两大类，各自又根据实际情况扩展成具体的计算方法，如图 1-5 所示。

- 流场数值解法
  - 分离解法
    - 非原始变量法
      - 涡量-流函数法
      - 涡量-速度法
    - 原始变量法
      - 压力修正法
      - 解压力泊松方程法
      - 人为压缩法
  - 耦合解法
    - 所有变量全场联立求解
    - 部分变量全场联立求解
    - 局部地区所有变量联立求解

图 1-5 流场数值计算方法的分类

1. 分离解法

分离解法不直接求解联立方程组，而是顺序地、逐个地求解各变量代数方程组。分离解法中应用广泛的是压力修正法，其求解基本过程如下。

(1) 假定初始压力场。

(2) 利用压力场求解动量方程，得到速度场。

(3) 利用速度场求解连续方程，使压力场得到修正。

(4) 根据需要，求解湍流方程及其他标量方程。

(5) 判断当前时间步上的计算是否收敛。若不收敛，返回第二步，迭代计算；若收敛，重复上述步骤，计算下一时间步的物理量。

压力修正法有很多实现方式，其中，压力耦合方程组的半隐式方法(SIMPLE 算法)应用最为广泛，也是各种商用 CFD 软件普遍采纳的算法。

2. 耦合解法

耦合解法同时求解离散方程组，联立求解出各变量(等)，其求解过程如下。

(1) 假定初始压力和速度等变量，确定离散方程的系数及常数项等。

(2) 联立求解连续方程、动量方程、能量方程。

(3) 求解湍流方程及其他标量方程。

(4) 判断当前时间步上的计算是否收敛。若不收敛，返回第二步，迭代计算；若收敛，重复上述步骤，计算下一时间步的物理量。

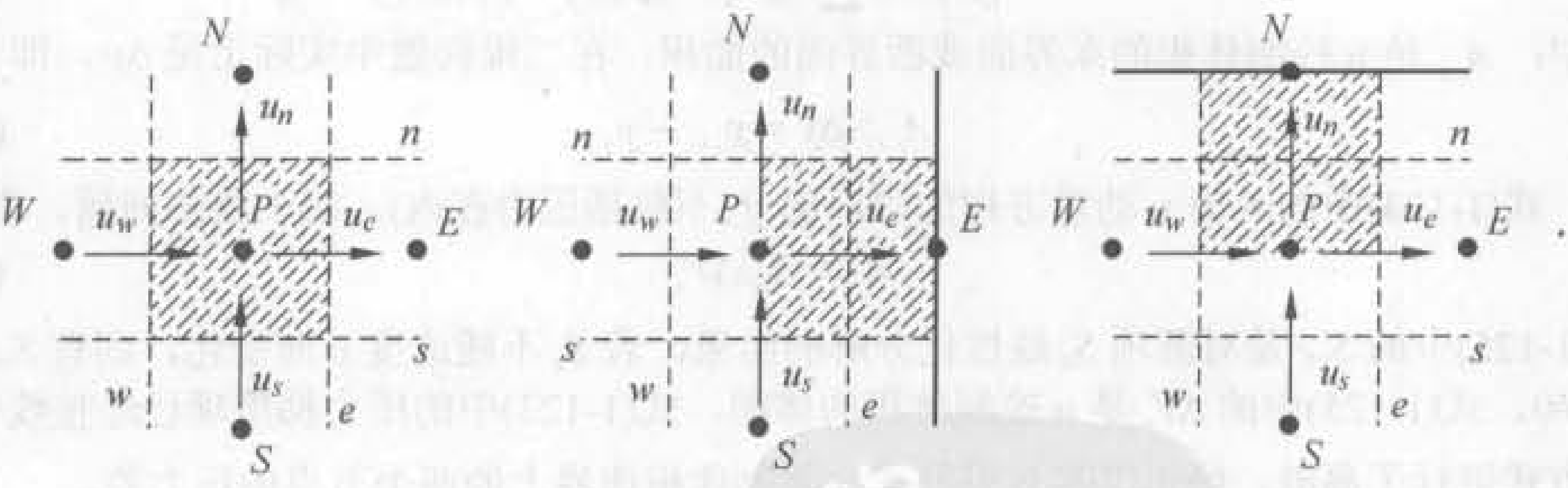
1.4.1 SIMPLE 算法详解

1. 交错网格

使用交错网格，主要是为了解决在普通网格上离散控制方程时不能检测有问题的压力场的问题，同时，交错网格也是SIMPLE算法实现的基础。

所谓交错网格(staggered grid)，就是将标量(如压力 p、温度 T 和密度等)在正常的网格节点上存储和计算，而将速度的各分量分别在错位后的网格上存储和计算，错位后网格的中心位于原控制体积的界面上。这样对于二维问题，就有三套不同的网格系统，分别用于存储 p、u 和 v；而对于三维问题，就有四套网格系统，分别用于存储 p、u、v、w。

二维流动计算的交错网格系统如图1-6所示，主控制体积为求解压力 p 的控制体积，称为标量控制体积(scale control volume)或 p 控制体积，控制体积的节点 P 称为主节点或标量节点(scale node)(如图1-6(a)所示)。速度 u 在主控制体积的东、西界面 e 和 w 上定义和存储，速度 v 在主控制体积的南、北界面 s 和 n 上定义和存储。u 和 v 各自的控制体积则是分别以速度所在位置(界面 e 和界面 n)为中心的，分别称为 u 控制体积(u-control volume)和 v 控制体积(v-control volume)，如图1-6(b)和(c)所示。可以看到，u 控制体积和 v 控制体积是与主控制体积不一致的，u 控制体积与主控制体积在 x 方向上有半个网格步长的错位，而 v 控制体积与主控制体积则在 y 方向上有半个步长的错位。



(a) 主控制体积　(b) u 控制体积　(c) v 控制体积

图1-6　控制体积

图1-7中所示的均匀网格是向后错位的，因为 u 的速度 $u_{i,J}$ 的 i 位置到标量节点(I,J)的是$-1/2\ \delta x_u$；同样，v 速度 $v_{I,j}$ 的 j 位置到标量节点(I,J)的距离是$-1/2\delta y_v$。当然，也可以选用向后两个错位的速度网格。

在使用了上述交错网格后，生成离散方程的方法和过程与原来的基于普通网格的方法和过程是一样的，只是需要注意所使用的控制体积有所变化。在交错网格中，由于所有标量(如压力、温度、密度等)仍然在主控制体积上存储，因此，以这些标量为因变量的输运

方程的离散过程及离散结果与前面的一样，主要是在交错网格中生成的 u 和 v 两个动量方程的离散方程时，积分用的控制体积不再是原来的主控制体积，而是 u 和 v 各自的控制体积，同时压力梯度项从源项中分离出来。例如，对 u 控制体积，该项积分为 $\int_{y_j}^{y_{j+1}}\int_{x_{i-1}}^{x_i}\left(-\frac{\partial p}{\partial x}\right)\mathrm{d}x\mathrm{d}y=(p_{I-1,J}-p_{I,J})A_{i,J}$。从而，对于方程(1-98)的离散方程，对 u 方向的动量方程使用 u 控制体积，可以写出在位置(i,J)处的关于速度 $u_{i,J}$ 的动量方程的离散形式：

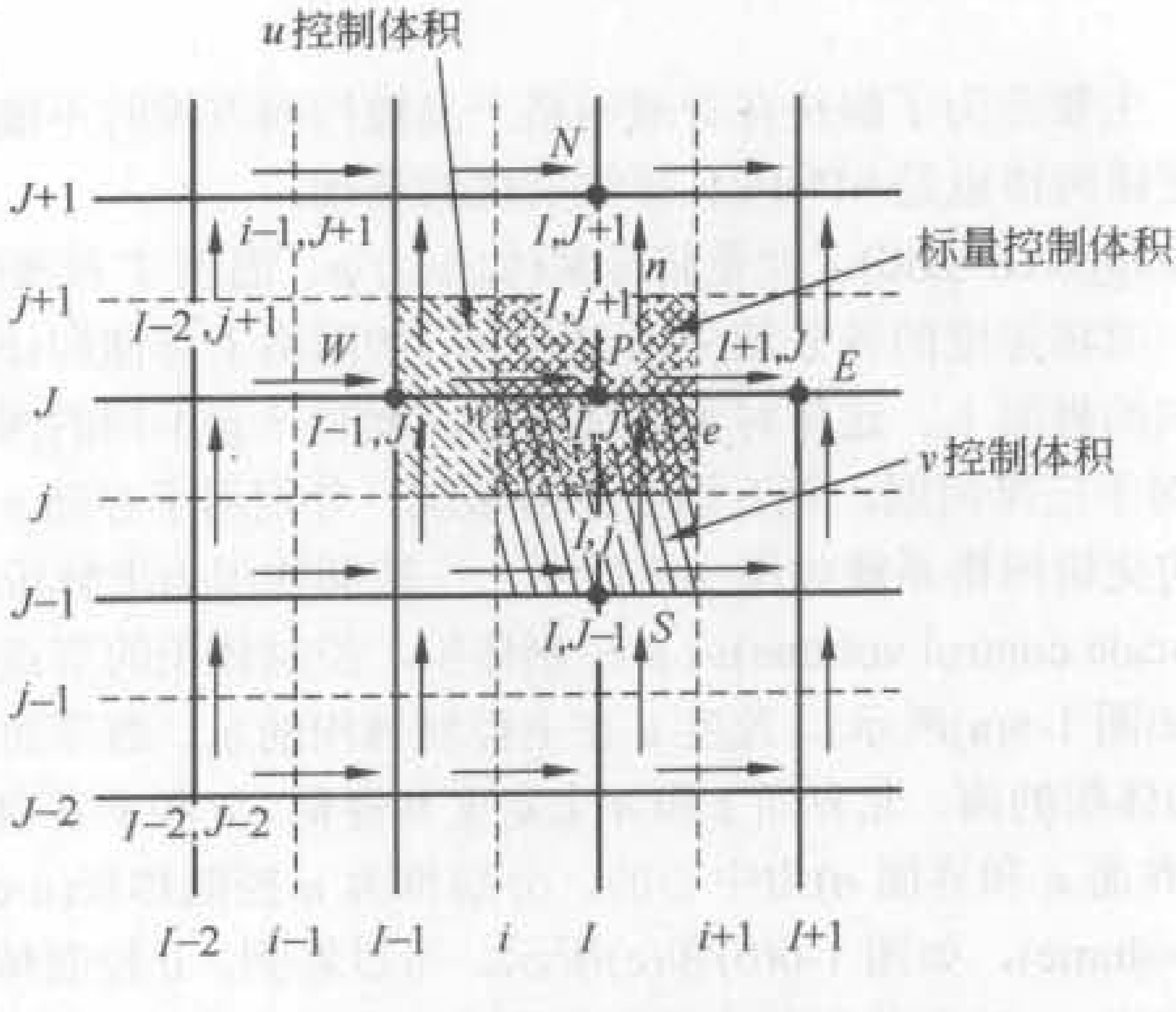


图 1-7 交错网格

$$a_{i,J}u_{i,J}=\sum a_{nb}u_{nb}+(p_{I-1,J}-p_{I,J})A_{i,J}+b_{i,J} \tag{1-123}$$

式中：$A_{i,J}$ 是 u 控制体积的东界面或西界面的面积，在二维问题中实际上是 Δy，即

$$A_{i,J}=\Delta y=y_{j+1}-y_j \tag{1-124}$$

式(1-123)中的 b 为 u 动量方程的源项部分(不包括压力在内)。对于稳态问题，有

$$b_{i,J}=S_{uC}\Delta V_u \tag{1-125}$$

式(1-125)中的 S_{uC} 是对源项 S_u 线性化分解的结果，若 S_u 不随速度 u 而变化，则有 $S_{uC}\geqslant S_u$，S_{uP}=0。式(1-125)中的 ΔV_u 是 u 控制体积的体积。式(1-123)中的压力梯度项已经按线性插值的方式进行了离散，线性插值时使用了 u 控制体积边界上的两个节点的压力差。

在求和记号 $\sum a_{nb}u_{nb}$ 中所包含的 E、W、N 和 S 四个邻点是$(i-1,J)$,$(i+1,J)$,$(i,J+1)$和$(i,J+1)$，它们的位置及主速度在图 1-8 中标出。图中阴影部分是 u 控制体积。图 1-8 中的 u 控制体积与图 1-7 中是一致的，这可从节点的编号看出。但是，图 1-8 中 u 控制体积的中心也用 P 来标记，其界面点也用 e、w、n 和 s 来标记，这里的标记与图 1-7 中的同名标记及系数 a 是由式(1-98)给定的。即

$$a_{i,J}=\sum a_{nb}+\Delta F-S_{uP}\Delta V_u \tag{1-126}$$

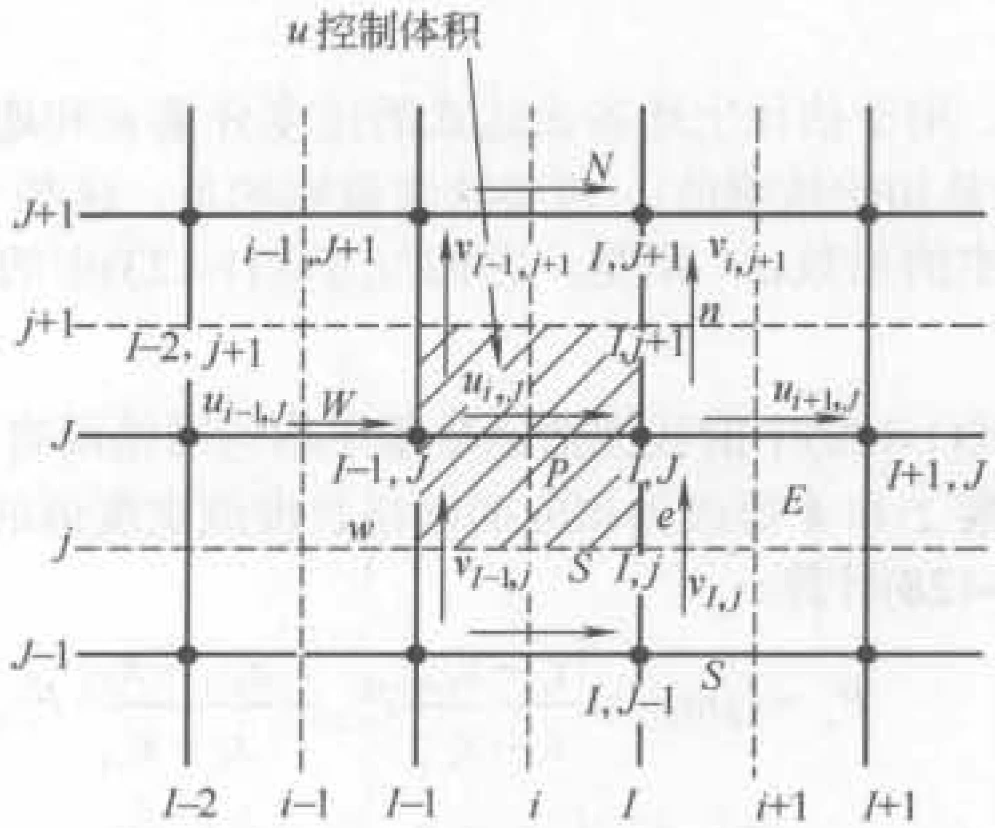


图 1-8　u 控制体积及其邻点的速度分量

系数 a_{nb} 取决于所采用的离散格式，在计算式中含有 u 控制体积界面上的对流质量流量 F 与扩散传导性 D、在采用新编号系统下的计算公式为

$$F_w=(\rho u)_w=\frac{F_{i,J}+F_{i-1,J}}{2}=\frac{1}{2}\left(\frac{\rho_{I,J}+\rho_{I-1,J}}{2}u_{i,J}+\frac{\rho_{I-1,J}+\rho_{I-2,J}}{2}u_{i-1,J}\right) \quad (1\text{-}127\text{a})$$

$$F_e=(\rho u)_e=\frac{F_{i+1,J}+F_{i,J}}{2}=\frac{1}{2}\left(\frac{\rho_{I+1,J}+\rho_{I,J}}{2}u_{i+1,J}+\frac{\rho_{I,J}+\rho_{I-1,J}}{2}u_{i,J}\right) \quad (1\text{-}127\text{b})$$

$$F_s=(\rho u)_s=\frac{F_{I,j}+F_{I-1,j}}{2}=\frac{1}{2}\left(\frac{\rho_{I,J}+\rho_{I,J-1}}{2}v_{I,j}+\frac{\rho_{I-1,J}+\rho_{I-1,J-1}}{2}v_{I-1,j}\right) \quad (1\text{-}127\text{c})$$

$$F_n=(\rho u)_n=\frac{F_{I,j+1}+F_{I-1,j+1}}{2}=\frac{1}{2}\left(\frac{\rho_{I,J+1}+\rho_{I,J}}{2}v_{I,j+1}+\frac{\rho_{I-1,J+1}+\rho_{I-1,J}}{2}v_{I-1,j+1}\right) \quad (1\text{-}127\text{d})$$

$$D_w=\frac{\Gamma_{I-1,J}}{x_i-x_{i-1}} \quad (1\text{-}127\text{e})$$

$$D_e=\frac{\Gamma_{I,J}}{x_{i+1}-x_i} \quad (1\text{-}127\text{f})$$

$$D_s=\frac{\Gamma_{I-1,J}+\Gamma_{I,J}+\Gamma_{I-1,J-1}+\Gamma_{I,J-1}}{4(y_J-y_{J-1})} \quad (1\text{-}127\text{g})$$

$$D_n=\frac{\Gamma_{I-1,J+1}+\Gamma_{I,J+1}+\Gamma_{I-1,J}+\Gamma_{I,J}}{4(y_{J+1}-y_J)} \quad (1\text{-}127\text{h})$$

采用交错网格对动量方程离散时，涉及不同类别的控制体积，不同的物理量分别在各自相应的控制体积的节点上定义和存储。例如，密度是在标量控制体积的节点上存储的，如图 1-8 中的标量节点(I,J)；而速度分量却是在错位后的速度控制体积的节点上存储的，如图 1-8 中的速度节点(i,J)。这样就会出现这种情况：在速度节点处不存在密度值，而在标量节点处找不到速度值，当在某个确定位置处的某个复合物理量(如式(1-127)中的流通量 F)同时需要该处的密度及速度时，要么找不到该处的密度，要么找不到该处的速度。为此，需要在计算过程中通过插值来解决。式(1-127)表明，标量(密度)及速度分量在 u 控制体积的界面上是不存在的，这时，根据周边的最近邻点的信息，使用二点或四点平均的办法来

处理。

在每次迭代过程中，用于估计上述各表达式的速度分量 u 和速度分量 v 是上一次迭代后的数值(在首次迭代时是初始猜测值)。需要特殊说明的是，这些“已知的”速度值 u 和 v 也用于计算方程(1-123)中的系数 a，但是，它们与方程(1-123)中的待求 $u_{i,J}$ 和 u_{nb} 是完全不同的。

还需要说明的是，式(1-127)中的线性插值是基于均匀网格而言的，若网格是不均匀的，应该将式(1-127)中的系数 2 和 4 等改为相应的网格长度或宽度值的组合。例如，对于不均匀网格上的 F_w，按式(1-128)计算：

$$F_w=(\rho u)_w=\frac{x_i-x_{I-1}}{x_i-x_{i-1}}F_{i,J}+\frac{x_{I-1}-x_{i-1}}{x_i-x_{i-1}}F_{I-1,J}$$

$$=\frac{x_i-x_{I-1}}{x_i-x_{i-1}}\left(\frac{x_I-x_i}{x_I-x_{I-1}}\rho_{I,J}+\frac{x_i-x_{I-1}}{x_I-x_{I-1}}\rho_{I-1,J}\right)u_{i,J}+ \tag{1-128}$$

$$\frac{x_{I-1}-x_{i-1}}{x_i-x_{i-1}}\left(\frac{x_{I-1}-x_i}{x_{I-1}-x_{I-2}}\rho_{I-1,J}+\frac{x_{i-1}-x_{I-2}}{x_{I-1}-x_{I-2}}\rho_{I-2,J}\right)u_{i-1,J}$$

按上述同样的方式，可以写出在新的编号系统中，对于在位置(I,j)处的关于速度 $v_{I,j}$ 的离散动量方程：

$$a_{I,j}v_{I,j}=\sum a_{nb}v_{nb}+(p_{I,J-1}-p_{I,J})A_{I,j}+b_{I,j} \tag{1-129}$$

建立式(1-129)所使用的 v 控制体积表示在图 1-9 中。

在求和记号 $\sum a_{nb}u_{nb}$ 中所包含的 4 个邻点及其主速度也在图 1-9 中标出。在系数 $a_{I,j}$ 和 a_{nb} 中，同样包含在 v 控制体积界面上的对流质量流量 F 与扩散传导性 D，计算公式如下：

$$F_w=(\rho u)_w=\frac{F_{i,J}+F_{i,J-1}}{2}=\frac{1}{2}\left(\frac{\rho_{I,J}+\rho_{I-1,J}}{2}u_{i,J}+\frac{\rho_{I-1,J-1}+\rho_{I,J-1}}{2}u_{i,J-1}\right) \tag{1-130a}$$

$$F_e=(\rho u)_e=\frac{F_{i+1,J}+F_{i+1,J-1}}{2}=\frac{1}{2}\left(\frac{\rho_{I+1,J}+\rho_{I,J}}{2}u_{i+1,J}+\frac{\rho_{I,J-1}+\rho_{I+1,J-1}}{2}u_{i+1,J+1}\right) \tag{1-130b}$$

$$F_s=(\rho v)_s=\frac{F_{I,j-1}+F_{I,j}}{2}=\frac{1}{2}\left(\frac{\rho_{I,J-1}+\rho_{I,J-2}}{2}v_{I,j-1}+\frac{\rho_{I,J}+\rho_{I,J-1}}{2}v_{I-1,j}\right) \tag{1-130c}$$

$$F_n=(\rho v)_n=\frac{F_{I,j}+F_{I,j+1}}{2}=\frac{1}{2}\left(\frac{\rho_{I,J}+\rho_{I,J-1}}{2}v_{I,j}+\frac{\rho_{I,J+1}+\rho_{I,J}}{2}v_{I,j+1}\right) \tag{1-130d}$$

$$D_w=\frac{\Gamma_{I-1,J-1}+\Gamma_{I,J-1}+\Gamma_{I-1,J}+\Gamma_{I,J}}{4(x_I-x_{I-1})} \tag{1-130e}$$

$$D_e=\frac{\Gamma_{I,J-1}+\Gamma_{I+1,J-1}+\Gamma_{I,J}+\Gamma_{I+1,J}}{4(x_{I+1}-x_I)} \tag{1-130f}$$

$$D_s=\frac{\Gamma_{I,J-1}}{y_j-y_{j-1}} \tag{1-130g}$$

$$D_n=\frac{\Gamma_{I,J}}{y_{j+1}-x_j} \tag{1-130h}$$

同样，在每个迭代层次上，用于估计上述各表达式的速度分量 u 和速度分量 v 均取上一次迭代后的数值(在首次迭代时是初始猜测值)。

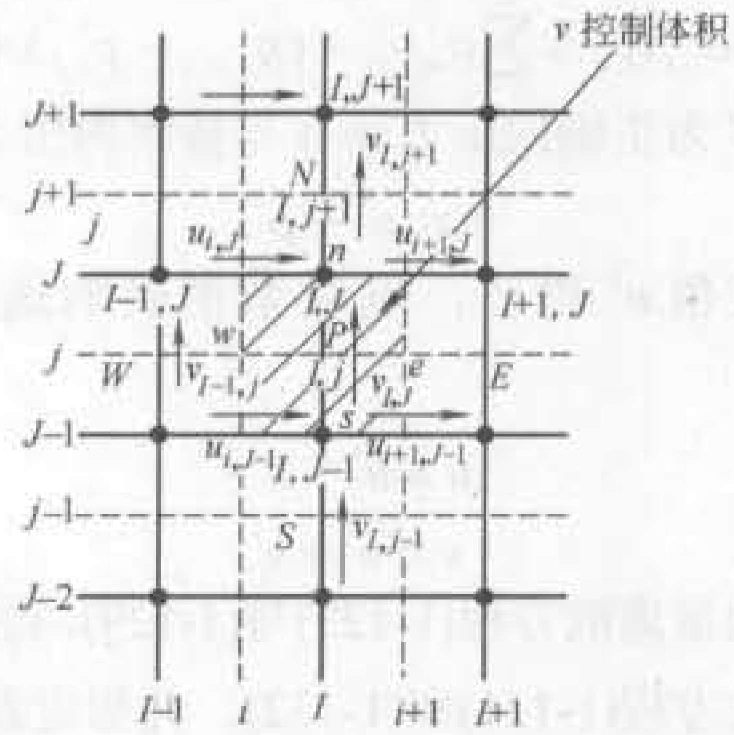


图 1-9 v 控制体积及其邻点的速度分量

给定一个压力场 p，便可针对每个 u 控制体积和 v 控制体积写出式(1-123)和式(1-129)所示的动量方程的离散方程，并可以从中求解出速度场。如果压力场是正确的，所得到的速度场将满足连续方程。

2. SIMPLE 算法

SIMPLE 算法是目前工程上应用最为广泛的一种流场计算方法，它属于压力修正法的一种。

SIMPLE 是英文 semi-implicit method for pressure-linked equations 的缩写，意为“求解压力耦合方程组的半隐式方法”。该方法由 Patankar 与 Spalding 于 1972 年提出，是一种主要用于求解不可压流场的数值方法(也可用于求解可压流动)。它的核心是采用“猜测-修正”的过程，在交错网格的基础上来计算压力场，从而达到求解动量方程(Navier-Stokes 方程)的目的。

SIMPLE 算法的基本思想可描述如下：对于给定的压力场(它可以是假定的值，或是上一次迭代计算所得到的结果)，求解离散形式的动量方程，得出速度场。因为压力场是假定的或不精确的，这样，由此得到的速度场一般不满足连续方程，因此，必须对给定的压力场加以修正。修正的原则是：与修正后的压力场相对应的速度场能满足这一迭代层次上的连续方程。据此原则，我们把由动量方程的离散形式所规定的压力与速度的关系代入连续方程的离散形式，从而得到压力修正方程，由压力修正方程得出压力修正值；接着，根据修正后的压力场，求得新的速度场；然后检查速度场是否收敛；若不收敛，用修正后的压力值作为给定的压力场，开始下一层次的计算；如此反复，直到获得收敛的解。

在上述求解过程中，如何获得压力修正值(即如何构造压力修正方程)，以及如何根据压力修正值确定“正确”的速度(即如何构造速度修正方程)，是 SIMPLE 算法的两个关键问题。为此，下面先解决这两个问题，然后给出 SIMPLE 算法的求解步骤。

1) 速度修正方程

考察一直角坐标系下的二维层流稳态问题。设有初始的猜测压力场 p^*，我们知道，动量方程的离散方程可借助该压力场得以求解，从而求出相应的速度分量 u^* 和 v^*。

根据动量方程的离散方程(1-123)和(1-129)，有

$$a_{I,J}u_{I,J}^* = \sum a_{nb}u_{nb}^* + (p_{I-1,J}^* - p_{I,J}^*)A_{I,J} + b_{I,J} \tag{1-131}$$

$$a_{I,J}v_{I,J}^* = \sum a_{nb}v_{nb}^* + (p_{I,J-1}^* - p_{I,J}^*)A_{I,J} + b_{I,J} \tag{1-132}$$

现在，定义压力修正值 p' 为正确的压力场 p 与猜测的压力场 p^* 之差，有

$$p = p^* + p' \tag{1-133}$$

同样地，定义速度修正值 u' 和 v'，以联系正确的速度场 (u,v) 与猜测的速度场 (u^*,v^*)，有

$$u = u^* + u' \tag{1-134}$$

$$v = v^* + v' \tag{1-135}$$

将正确的压力场 p 代入动量离散方程(1-123)与(1-129)，得到正确的速度场 (u,v)。现在，从方程(1-123)和(1-129)中减去方程(1-131)和(1-132)，并假定源项 b 不变，有

$$a_{i,J}(u_{i,J} - u_{i,J}^*) = \sum a_{nb}(u_{nb} - u_{nb}^*) + [(p_{I-1,J} - p_{I-1,J}^*) - (p_{I,J} - p_{I,J}^*)]A_{i,J} \tag{1-136}$$

$$a_{I,j}(v_{I,j} - v_{I,j}^*) = \sum a_{nb}(v_{nb} - v_{nb}^*) + [(p_{I,J-1} - p_{I,J-1}^*) - (p_{I,J} - p_{I,J}^*)]A_{I,j} \tag{1-137}$$

引入压力修正值与速度修正值的表达式(1-133)～式(1-135)，方程(1-136)和(1-137)可写成

$$a_{i,J}u'_{i,J} = \sum a_{nb}u'_{nb} + (p'_{I-1,J} - p'_{I,J})A_{i,J} \tag{1-138}$$

$$a_{I,j}v'_{I,j} = \sum a_{nb}v'_{nb} + (p'_{I,J-1} - p'_{I,J})A_{I,j} \tag{1-139}$$

可以看出，由压力修正值 p' 可求出速度修正值 (u',v')。式(1-139)还表明，任一点上速度的修正值由两部分组成：一部分是与该速度在同一方向上的相邻两节点间压力修正值之差，这是产生速度修正值的直接的动力；另一部分是由邻点速度的修正值所引起的，这又可以视为四周压力的修正值对所讨论位置上速度改进的间接影响。

为了简化式(1-138)和(1-139)的求解过程，在此，引入如下近似处理：略去方程中与速度修正值相关的 $\sum a_{nb}u'_{nb}$ 和 $\sum a_{nb}v'_{nb}$。该近似是 SIMPLE 算法的重要特征。略去后的影响将在下一节要介绍的 SIMPLEC 算法中讨论。于是有

$$u'_{i,J} = d_{i,J}(p'_{I-1,J} - p'_{I,J}) \tag{1-140}$$

$$v'_{I,j} = d_{I,j}(p'_{I,J-1} - p'_{I,J}) \tag{1-141}$$

以上两种中，

$$d_{i,J} = \frac{A_{i,J}}{a_{i,J}},\ d_{I,j} = \frac{A_{I,j}}{a_{I,j}} \tag{1-142}$$

将式(1-140)和式(1-141)所描述的速度修正值，代入式(1-134)和式(1-135)，有

$$u_{i,J} = u_{i,J}^* + d_{i,J}(p'_{I-1,J} - p'_{I,J}) \tag{1-143}$$

$$v_{I,j} = v_{I,j}^* + d_{I,j}(p'_{I,J-1} - p'_{I,J}) \tag{1-144}$$

对于 $u_{i+1,J}$ 和 $v_{I,j-1}$，存在类似的表达式：

$$u_{i+1,J} = u_{i+1,J}^* + d_{i+1,J}(p'_{I,J} - p'_{I+1,J}) \tag{1-145}$$

$$v_{I,j+1} = v_{I,j+1}^* + d_{I,j+1}(p'_{I,J} - p'_{I,J+1}) \tag{1-146}$$

以上两式中：

$$d_{i+1,J} = \frac{A_{i+1,J}}{a_{i+1,J}},\ d_{I,j+1} = \frac{A_{I,j+1}}{a_{I,j+1}} \tag{1-147}$$

式(1-143)～式(1-147)表明，如果已知压力修正值 p'，便可对猜测的速度场 (u^*,v^*) 作出相应的速度修正，得到正确的速度场 (u,v)。

2)　压力修正方程

在上面的推导中，只考虑了动量方程，其实，如前所述，速度场还受连续方程(1-38)的约束。按本节开头的约定，这里暂不讨论瞬态问题。对于稳态问题，连续方程可写为

$$\frac{\partial(\rho u)}{\partial x}+\frac{\partial(\rho v)}{\partial y}=0 \tag{1-148}$$

针对如图 1-10 所示的标量控制体积，连续方程(1-148)满足如下离散形式：

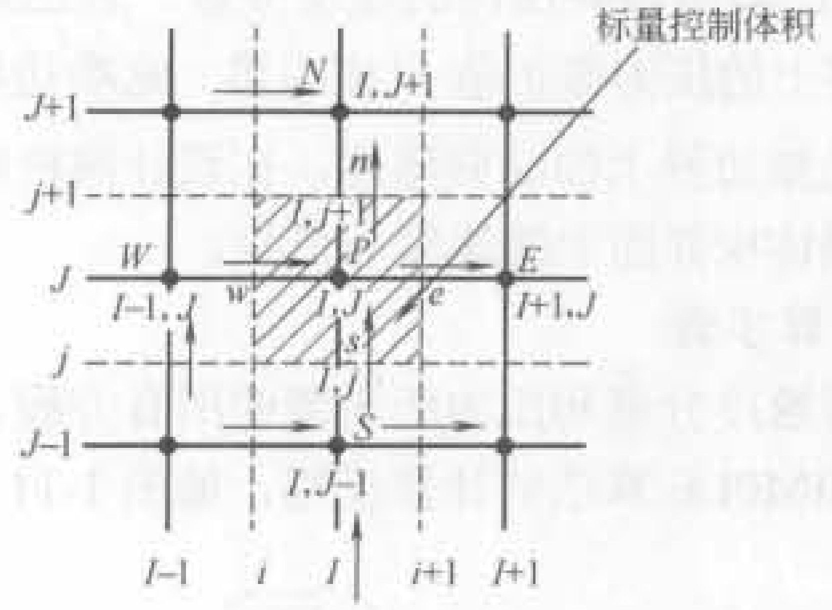


图 1-10　离散连续方程的标量控制体积

$$[(\rho uA)_{i+1,J}-(\rho uA)_{i,J}]+[(\rho vA)_{I,j+1}-(\rho vA)_{I,j}]=0 \tag{1-149}$$

将正确的速度值，即式(1-143)～式(1-147)，代入连续方程的离散方程(1-149)，有

$$\begin{aligned}&\{\rho_{i+1,J}A_{i+1,J}[u^*_{i+1,J}+d_{i+1,J}(p'_{I,J}-p'_{I+1,J})]-\rho_{i,J}A_{i,J}[u^*_{i,J}+d_{i,J}(p'_{I-1,J}-p'_{I,J})]\}+\\&\{\rho_{I,j+1}A_{I,j+1}[v^*_{I,j+1}+d_{I,j+1}(p'_{I,J}-p'_{I,J+1})]-\rho_{I,j}A_{I,j}[v^*_{I,j}+d_{I,j}(p'_{I,J-1}-p'_{I,J})]\}=0\end{aligned} \tag{1-150}$$

整理后，得

$$\begin{aligned}&[(\rho dA)_{i+1,J}+(\rho dA)_{i,J}+(\rho dA)_{I,j+1}+(\rho dA)_{I,j}]p'_{I,J}\\&=(\rho dA)_{i+1,J}p'_{I+1,J}+(\rho dA)_{i,J}p'_{I-1,J}+(\rho dA)_{I,j+1}p'_{I,J+1}+(\rho dA)_{I,j}p'_{I,J-1}+\\&\quad[(\rho u^*A)_{i,J}-(\rho u^*A)_{i+1,J}+(\rho u^*A)_{I,j}-(\rho u^*A)_{I,j+1}]\end{aligned} \tag{1-151}$$

该式可简记为

$$a_{I,J}p'_{I,J}=a_{I+1,J}p'_{I+1,J}+a_{I-1,J}p'_{I-1,J}+a_{I,J+1}p'_{I,J+1}+a_{I,J-1}p'_{I,J-1}+b'_{I,J} \tag{1-152}$$

式中：

$$a_{I+1,J}=(\rho dA)_{i+1,J} \tag{1-153a}$$

$$a_{I-1,J}=(\rho dA)_{i,J} \tag{1-153b}$$

$$a_{I,J+1}=(\rho dA)_{I,j+1} \tag{1-153c}$$

$$a_{I,J-1}=(\rho dA)_{I,j} \tag{1-153d}$$

$$a_{I,J}=a_{I+1,J}+a_{I-1,J}+a_{I,J+1}+a_{I,J-1} \tag{1-153e}$$

$$b'_{I,J}=(\rho u^*A)_{i,J}-(\rho u^*A)_{i+1,J}+(\rho v^*A)_{I,j}-(\rho v^*A)_{I,j+1} \tag{1-153f}$$

式(1-152)表示连续方程的离散方程，即压力修正值 p' 的离散方程。方程中的源项 b' 是由于不正确的速度场 (u^*,v^*) 所导致的“连续性”不平衡量。通过求解方程(1-152)，可得到

空间所有位置的压力修正值 p'。

如同处理式(1-127)中的密度一样，式(1-153)中的 ρ 是标量控制体积界面上的密度值，同样需要通过插值得到，这是因为密度 ρ 是在标量控制体积中的节点(即控制体积的中心)定义和存储的，在标量控制体积界面上不存在可直接引用的值。无论采用何种插值方法，对于交界面所属的两个控制体积，必须采用同样的 ρ 值。

为了求解方程(1-152)，还必须对压力修正值的边界条件作出说明。实际上，压力修正方程是动量方程和连续方程的派生物，不是基本方程，故其边界条件也与动量方程的边界条件相联系。在一般的流场计算中，动量方程的边界条件通常有两类：第一，已知边界上的压力(速度未知)；第二，已知沿边界法向的速度分量。若已知边界压力 $\overline{p}$，可在该段边界上令 $p^*=\overline{p}$，则该段边界上的压力修正值 p' 应为零。这类边界条件类似于热传导问题中已知温度的边界条件。若已知边界上的法向速度，在设计网格时，最好令控制体积的界面与边界相一致，这样，控制体积界面上的速度为已知。

3) SIMPLE 算法的计算步骤

至此，已经得出了求解速度分量和压力所需要的所有方程。根据前面介绍的 SIMPLE 算法的基本思想，可给出 SIMPLE 算法的计算流程，如图 1-11 所示。

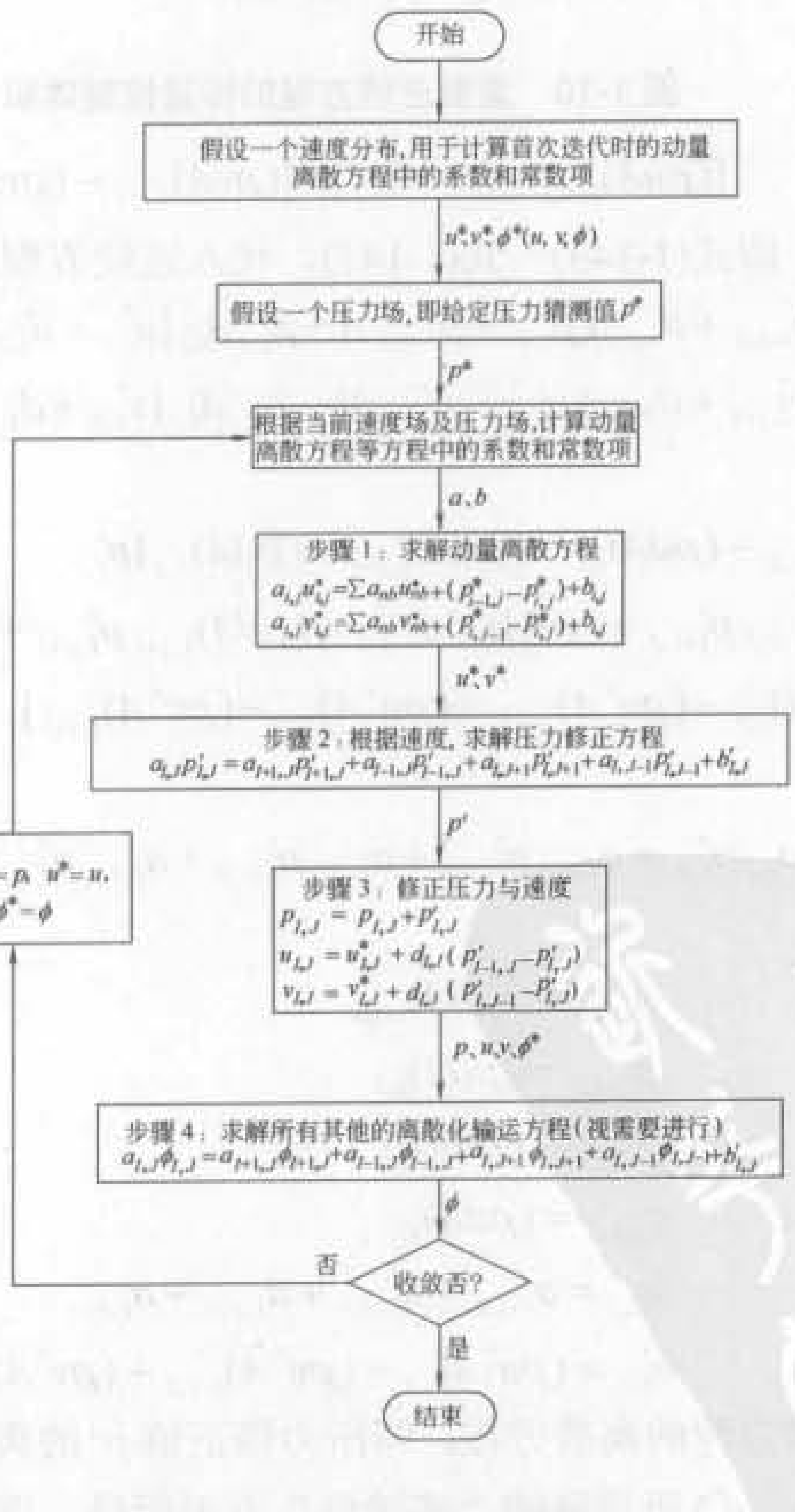


图 1-11 SIMPLE 算法流程图

1.4.2 其他算法介绍

SIMPLE 算法自问世以来，在被广泛应用的同时，也以不同方式不断地得到改善和发展，其中最著名的改进算法包括 SIMPLEC、SIMPLER 和 PISO。本节介绍这些改进算法，并对各算法作一对比。

1. SIMPLER 算法

SIMPLER 是英文 SIMPLE revised 的缩写，顾名思义是 SIMPLE 算法的改进版本。它是由 SIMPLER 算法的创始者之一 Patanker 完成的。

我们知道，在 SIMPLER 算法中，为了确定动量离散方程的系数，一开始就假定了一个速度分布，同时又独立地假定了一个压力分布，两者之间一般是不协调的，从而影响了迭代计算的收敛速度。实际上，不必在初始时刻单独假定一个压力场，因为与假定的速度场相协调的压力场是可以通过动量方程求出的。另外，在 SIMPLER 算法中对压力修正值 p' 采用了欠松弛处理，而松弛因子是比较难确定的，因此，速度场的改进与压力场的改进不能同步进行，最终影响收敛速度。于是，Patanker 便提出了这样的想法：p' 只用修正速度，压力场的改进则另谋更合适的方法。将上述两方面的思想结合起来，就构成了 SIMPLER 算法。

在 SIMPLER 算法中，经过离散后的连续方程(1-149)用于建立一个压力的离散方程，而不像在 SIMPLE 算法中用来建立压力修正方程。从而，可直接得到压力，而不需要修正。但是，速度仍需要通过 SIMPLE 算法中的修正方程即式(1-143)～式(1-147)来修正。

将离散后的动量方程式(1-123)和式(1-129)重新改写后，有

$$u_{i,j}=\frac{\sum a_{nb}u_{nb}+b_{i,j}}{a_{i,j}}+\frac{A_{i,j}}{a_{i,j}}(p_{i-1,j}-p_{i,j}) \tag{1-154}$$

$$v_{i,j}=\frac{\sum a_{nb}v_{nb}+b_{i,j}}{a_{i,j}}+\frac{A_{i,j}}{a_{i,j}}(p_{i,j-1}-p_{i,j}) \tag{1-155}$$

在 SIMPLER 算法中，定义伪速度 $\hat{u}$ 与 $\hat{v}$ 如下：

$$\hat{u}=\frac{\sum a_{nb}u_{nb}+b_{I,J}}{a_{I,J}} \tag{1-156}$$

$$\hat{v}=\frac{\sum a_{nb}v_{nb}+b_{I,J}}{a_{I,J}} \tag{1-157}$$

这样，式(1-154)与式(1-155)可写为

$$u_{i,j}=\hat{u}_{i,j}+d_{i,j}(p_{i-1,j}-p_{i,j}) \tag{1-158}$$

$$v_{i,j}=\hat{v}_{i,j}+d_{i,j}(p_{i,j-1}-p_{i,j}) \tag{1-159}$$

以上两式中的系数 d，仍沿用 1.4.1 节所给出的计算公式。同样可写出 $u_{i+1,j}$ 与 $v_{i,j+1}$ 的表达式。然后，将 $u_{i,j}$、$v_{i,j}$、$u_{i+1,j}$ 与 $v_{i,j+1}$ 的表达式代入离散后的连续方程(1-149)，有

$$\begin{aligned}&\left\{\rho_{i+1,j}A_{i+1,j}[\hat{u}_{i-1,j}+d_{i+1,j}(p_{i,j}-p_{i+1,j})]-\rho_{i,j}A_{i,j}[\hat{u}_{i,j}+d_{i,j}(p_{i-1,j}-p_{i,j})]\right\}+\\&\left\{\rho_{i,j+1}A_{i,j+1}[\hat{u}_{i,j+1}+d_{i,j+1}(p_{i,j}-p_{i,j+1})]-\rho_{i,j}A_{i,j}[\hat{v}_{i,j}+d_{i,j}(p_{i,j-1}-p_{i,j})]\right\}=0\end{aligned} \tag{1-160}$$

整理后，得到离散后的压力方程：

$$a_{I,J}p_{I,J} = a_{I+1,J}p_{I+1,J} + a_{I-1,J}p_{I-1,J} + a_{I,J+1}p_{I,J+1} + a_{I,J-1}p_{I,J-1} + b_{I,J} \quad (1\text{-}161)$$

式中：

$$a_{I+1,J} = (\rho dA)_{i+1,J} \quad (1\text{-}162a)$$

$$a_{I-1,J} = (\rho dA)_{i,J} \quad (1\text{-}162b)$$

$$a_{I,J+1} = (\rho dA)_{I,j+1} \quad (1\text{-}162c)$$

$$a_{I,J-1} = (\rho dA)_{I,j} \quad (1\text{-}162d)$$

$$a_{I,J} = a_{I+1,J} + a_{I-1,J} + a_{I,J+1} + a_{I,J-1} \quad (1\text{-}162e)$$

$$b_{I,J} = (\rho\hat{u}A)_{i,J} - (\rho\hat{u}A)_{i+1,J} + (\rho\hat{v}A)_{I,j} - (\rho\hat{v}A)_{I,j+1} \quad (1\text{-}162f)$$

我们注意到，方程(1-161)中的系数与压力修正方程(1-152)中的系数是一样的，差别仅在于源项 b。这里的源项 b 是用伪速度来计算的。因此，离散后的动量方程式(1-131)和式(1-132)，可借助上面得到的压力场来直接求解。这样，可求出速度分量 u^* 和 v^*。SIMPLER 算法流程图如图 1-12 所示。

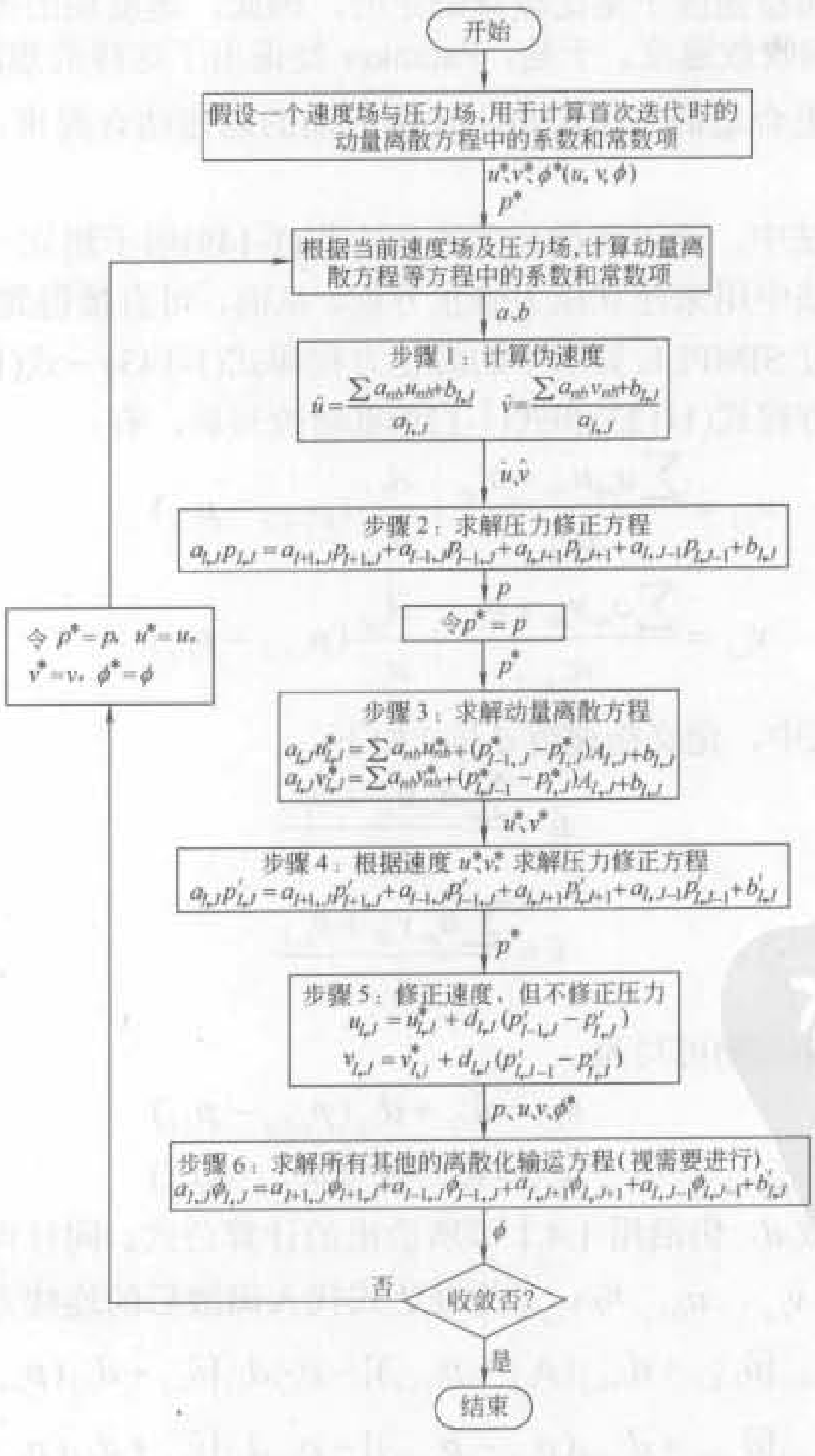


图 1-12 SIMPLER 算法流程图

在 SIMPLER 算法中，初始的压力场与速度场是协调的，且由 SIMPLER 算法算出的压力场不必作欠松弛处理，迭代计算时比较容易得到收敛解。但在 SIMPLER 的每一层迭代中，要比 SIMPLE 算法多解一个关于压力的方程组，一个迭代步内的计算量较大。总体而言，SIMPLER 的计算效率要高于 SIMPLE 算法。

2. SIMPLEC 算法

SIMPLEC 是英文 SIMPLE consistent 的缩写，意为协调一致的 SIMPLE 算法。它也是 SIMPLE 的改进算法之一，是由 Van Doormal 和 Raithby 提出的。

我们知道，在 SIMPLE 算法中，为求解的方便，略去了速度修正方程中的 $\sum a_{nb}u'_{nb}$ 项，从而把速度的修正完全归结为由于压差项的直接作用。这一作法虽然并不影响收敛解的值，但加重了修正值 p' 的负担，使得整个速度场迭代收敛速度降低。实际上，当我们在略去 $\sum a_{nb}u'_{nb}$ 时，犯了一个“不协调一致”的错误。为了能略去 $a_{nb}u'_{nb}$ 而同时又能使方程基本协调，试在 $u'_{i,j}$ 方程(1-138)的等号两端同时减去 $\sum a_{nb}u'_{i,j}$，有

$$(a_{i,j}-\sum a_{nb})u'_{i,j}=\sum a_{nb}(u'_{nb}-u'_{i,j})+A_{i,j}(p'_{i-1,j}-p'_{i,j}) \tag{1-163}$$

可以预期，$u'_{i,j}$ 与其邻点的修正值 u'_{nb} 具有相同的数量级，因而略去 $\sum a_{nb}(u'_{nb}-u'_{i,j})$ 所产生的影响远比在方程(1-138)中不计 $\sum a_{nb}u'_{nb}$ 所产生的影响要小得多。于是有

$$u'_{i,j}=d_{i,j}(p'_{i-1,j}-p'_{i,j}) \tag{1-164}$$

式中：

$$d_{i,j}=\frac{A_{i,j}}{(a_{i,j}-\sum a_{nb})} \tag{1-165}$$

类似地，有

$$v'_{i,j}=d_{i,j}(p'_{i,j-1}-p'_{i,j}) \tag{1-166}$$

式中：

$$d_{i,j}=\frac{A_{i,j}}{(a_{i,j}-\sum a_{nb})} \tag{1-167}$$

将式(1-166)和式(1-167)代入 SIMPLE 算法中的式(1-143)和式(1-144)，得到修正后的速度计算式：

$$u_{i,j}=u^*_{i,j}+d_{i,j}(p'_{i-1,j}-p'_{i,j}) \tag{1-168}$$

$$v_{i,j}=v^*_{i,j}+d_{i,j}(p'_{i,j-1}-p'_{i,j}) \tag{1-169}$$

式(1-168)和式(1-169)在形式上与式(1-143)和式(1-144)一致，只是其中的系数项 d 的计算公式不同，现在需要按式(1-165)和式(1-167)计算。

这就是 SIMPLEC 算法。SIMPLEC 算法与 SIMPLE 算法的计算步骤相同，只是速度修正方程中的系数项 d 的计算公式有所区别。

由于 SIMPLEC 算法没有像 SIMPLE 算法那样将 $\sum a_{nb}u'_{nb}$ 项忽略，因此，得到的压力修正值 p' 一般是比较合适的，因此，在 SIMPLEC 算法中可不再对 p' 进行欠松弛处理。但据作者的试验，适当选取一个稍小于 1 的 α_p 对 p' 进行欠松弛处理，对加快迭代过程中解的收敛也是有效的。

3. PISO 算法

PISO 是 pressure implicit with splitting of operators 的缩写，意为压力的隐式算子分割算法。PISO 算法是 Issa 于 1986 年提出的，起初是针对非稳态可压流动的无迭代计算所建立的一种压力速度计算程序，后来在稳态问题的迭代计算中也较广泛地使用了该算法。

PISO 算法与 SIMPLE、SIMPLEC 算法的不同之处在于：SIMPLE 和 SIMPLEC 算法是两步算法，即一步预测(图 1-11 中的步骤 1)和一步修正(图 1-11 中的步骤 2 和步骤 3)；而 PISO 算法增加了一个修正步，包含一个预测步和两个修正步，在完成了第一步修正得到 (u, v, p) 后寻求二次改进值，目的是使它们更好地同时满足动量方程和连续方程。PISO 算法由于使用了预测—修正—再修正三步，从而可加快单个迭代步中的收敛速度。现将三个步骤介绍如下。

1) 预测步

使用与 SIMPLE 算法相同的方法，利用猜测的压力场 p^*，求解动量离散方程(1-131)与方程式(1-132)，得到速度分量 u^* 与 v^*。

2) 第一步修正

所得到的速度场(u^*, v^*)一般不满足连续方程，除非压力场 p^* 是准确的。现引入对 SIMPLE 的第一个修正步，该修正步给出一个速度场(u^{**}, v^{**})，使其满足连续方程。此处的修正公式与 SIMPLE 算法中的式(1-140)和(1-141)完全一致，只不过考虑到在 PISO 算法还有第二个修正步，因此，使用不同的记法：

$$p^{**} = p^* + p' \tag{1-170}$$

$$u^{**} = u^* + u' \tag{1-171}$$

$$v^{**} = v^* + v' \tag{1-172}$$

这组公式用于定义修正后的速度 u^{**} 与 v^{**}：

$$u_{i,j}^{**} = u_{i,j}^* + d_{i,j}(p'_{i-1,j} - p'_{i,j}) \tag{1-173}$$

$$v_{i,j}^{**} = v_{i,j}^* + d_{i,j}(p'_{i,j-1} - p'_{i,j}) \tag{1-174}$$

就像在 SIMPLE 算法中一样，将式(1-173)与式(1-174)代入连续方程(1-149)，产生与式(1-151)具有相同系数和源项的压力修正方程。求解该方程，产生第一个压力修正值 p'。一旦压力修正值已知，可通过式(1-173)与式(1-174)获得速度分量 u^{**} 与 v^{**}。

3) 第二步修正

为了强化 SIMPLE 算法的计算，PISO 要进行第二步的修正。u^{**} 和 v^{**} 的动量离散方程是

$$a_{i,j}u_{i,j}^{**} = \sum a_{nb}u_{nb}^* + (p_{i-1,j}^{**} - p_{i,j}^{**})A_{i,j} + b_{i,j} \tag{1-175}$$

$$a_{i,j}v_{i,j}^{**} = \sum a_{nb}v_{nb}^* + (p_{i,j-1}^{**} - p_{i,j}^{**})A_{i,j} + b_{i,j} \tag{1-176}$$

注意这两式实际就是式(1-131)和式(1-132)。为引用方便，给出新的编号。

再次求解动量方程，可以得到两次修正的速度场(u^{***}, v^{***})：

$$a_{i,j}u_{i,j}^{***} = \sum a_{nb}u_{nb}^{**} + (p_{i-1,j}^{***} - p_{i,j}^{***})A_{i,j} + b_{i,j} \tag{1-177}$$

$$a_{i,j}v_{i,j}^{***} = \sum a_{nb}v_{nb}^{**} + (p_{i,j-1}^{***} - p_{i,j}^{***})A_{i,j} + b_{i,j} \tag{1-178}$$

注意修正步中的求和项是用速度分量 u^{**} 和 v^{**} 来计算的。

现在，从式(1-177)中减去式(1-175)，从式(1-178)中减去式(1-176)，有

$$u_{i,j}^{***}=u_{i,j}^{**}+\frac{\sum a_{nb}(u_{nb}^{**}-u_{nb}^{*})}{a_{i,j}}+d_{i,j}(p_{i-1,j}''-p_{i,j}'') \tag{1-179}$$

$$v_{i,j}^{***}=v_{i,j}^{**}+\frac{\sum a_{nb}(v_{nb}^{**}-v_{nb}^{*})}{a_{i,j}}+d_{i,j}(p_{i,j-1}''-p_{i,j}'') \tag{1-180}$$

以上两式中，记号 p'' 是压力的二次修正值。有了该记号，p^{***} 可表示为

$$p^{***}=p^{**}+p'' \tag{1-181}$$

将 u^{***} 和 v^{***} 的表达式(1-179)和(1-180)代入连续方程(1-149)，得到二次压力修正方程：

$$a_{i,j}p_{i,j}''=a_{i+1,j}p_{i+1,j}''+a_{i-1,j}p_{i-1,j}''+a_{i,j+1}p_{i,j+1}''+a_{i,j-1}p_{i,j-1}''+b_{i,j}'' \tag{1-182}$$

以上两式中，$a_{i,j}=a_{i+1,j}+a_{i-1,j}+a_{i,j+1}+a_{i,j-1}$。读者可参考建立方程(1-152)同样的过程，写出各系数如下：

$$a_{i+1,j}=(\rho dA)_{i+1,j} \tag{1-183a}$$

$$a_{i-1,j}=(\rho dA)_{i,j} \tag{1-183b}$$

$$a_{i,j+1}=(\rho dA)_{i,j+1} \tag{1-183c}$$

$$a_{i,j-1}=(\rho dA)_{i,j} \tag{1-183d}$$

$$b_{I,J}'=\left(\frac{\rho A}{a}\right)_{i,J}\sum a_{nb}(u_{nb}^{**}-u_{nb}^{*})-\left(\frac{\rho A}{a}\right)_{i+1,J}\sum a_{nb}(u_{nb}^{**}-u_{nb}^{*})+ \tag{1-183e}$$

$$\left(\frac{\rho A}{a}\right)_{I,j}\sum a_{nb}(v_{nb}^{**}-v_{nb}^{*})-\left(\frac{\rho A}{a}\right)_{I,j+1}\sum a_{nb}(v_{nb}^{**}-v_{nb}^{*})$$

下面对源项 b' 为何是式(1-183e)的形式作一简要分析和解释。

对比建立方程(1-152)的过程，可以发现，式(1-183e)中的各项，是因在 u^{***} 和 v^{***} 的表达式(1-179)和式(1-180)中存在 $\frac{\sum a_{nb}(u_{nb}^{**}-u_{nb}^{*})}{a_{i,j}}$ 和 $\frac{\sum a_{nb}(v_{nb}^{**}-v_{nb}^{*})}{a_{i,j}}$ 项所导致的，而在 u 和 v 的表达式(1-142)和式(1-143)中没有这样的项，因此，式(1-152)不存在类似式(1-183e)中的各项。但式(1-152)存在另外一个源项，即 $[(\rho u^{*}A)_{i,j}-(\rho u^{*}A)_{i+1,j}+(\rho v^{*}A)_{i,j}-(\rho v^{*}A)_{i,j+1}]$，这是因速度 u 和 v 的表达式(1-143)和式(1-144)中的 u^{*} 与 v^{*} 项所导致的。按此推断，在式(1-183e)中也应该存在类似表达式 $[(\rho u^{**}A)_{i,j}-(\rho u^{**}A)_{i+1,j}+(\rho v^{**}A)_{i,j}-(\rho v^{**}A)_{i,j+1}]$。但是，由于 u^{**} 和 v^{**} 满足连续方程，因此 $\left[(\rho u^{**}A)_{i,j}-(\rho u^{**}A)_{i+1,j}+(\rho v^{**}A)_{i,j}-(\rho v^{**}A)_{i,j+1}\right]$ 为 0。

现在，求解方程(1-182)，就可得到二次压力修正值 p''。这样，通过式(1-184)就可得到二次修正的压力场：

$$p^{***}=p^{**}+p''=p^{*}+p'+p'' \tag{1-184}$$

最后，求解方程(1-179)与(1-180)，得到二次修正的速度场。

在瞬态问题的非迭代计算中，压力场 p^{***} 与速度场(u^{***}, v^{***})被认为是准确的。对于稳态流动的迭代计算，PISO 算法的实施过程如图 1-13 所示。

PISO 算法要两次求解压力修正方程，因此，它需要额外的存储空间来计算二次压力修正方程中的源项。尽管该方法涉及较多的计算，但对比发现，它的计算速度很快，总体效率比较高。Fluent 的用户手册[7]推荐，对于瞬态问题，PISO 算法有明显的优势；而对于稳

态问题，可能选 SIMPLE 或 SIMPLEC 算法更合适。

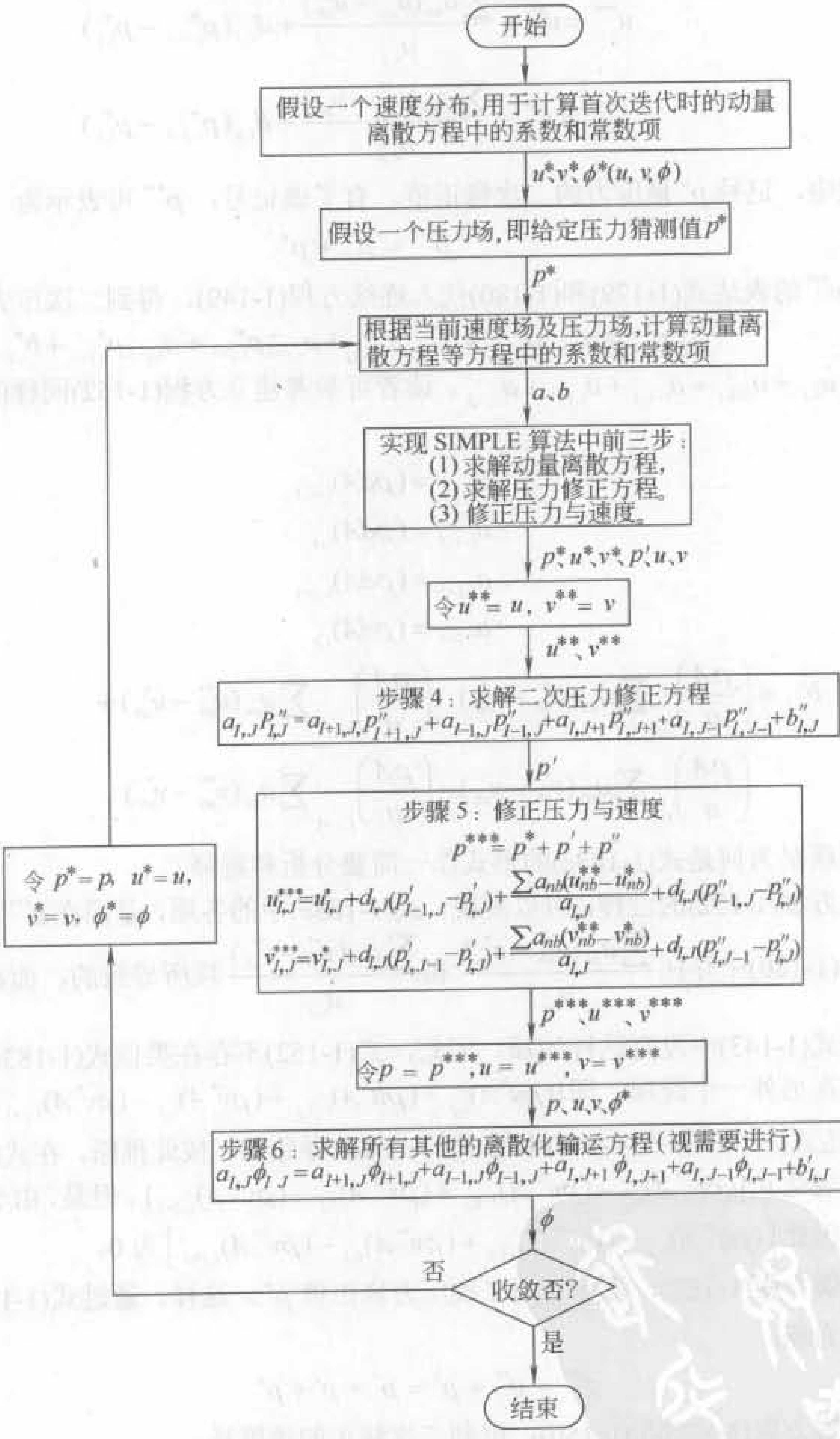


图 1-13 PISO 算法流程图

4. SIMPLE 系列算法的比较

SIMPLE 算法是 SIMPLE 系列算法的基础，目前在各种 CFD 软件中均提供这种算法。SIMPLE 的各种改进算法，主要是提高了计算的收敛性，从而可缩短计算时间。

在 SIMPLE 算法中，压力修正值 p' 能够很好地满足速度修正的要求，但对压力修正不是十分理想。改进后的 SIMPLER 算法只用压力修正值 p' 来修正速度，另外构建一个更加有效的压力方程来产生“正确”的压力场。由于在推导 SIMPLER 算法的离散化压力方程时，没有任何项被忽略，因此所得到的压力场与速度场相适应。在 SIMPLER 算法中，正确的速度场将导致正确的压力场，而在 SIMPLE 算法中则不是这样。所以 SIMPLER 算法是在很高的效率下正确计算压力场的，这一点在求解动量方程时有明显优势。虽然 SIMPLER 算法的计算量比 SIMPLE 算法高出 30%左右，但其较快的收敛速度使得计算时间减少 30%～50%。

SIMPLEC 算法和 PISO 算法总体上与 SIMPLER 算法具有同样的计算效率，相互之间很难区分谁高谁低，对于不同类型的问题每种算法都有自己的优势。一般来讲，动量方程与标量方程(如温度方程)如果不是耦合在一起的，则 PISO 算法在收敛性方面显得很健壮，且效率较高。而在动量方程与标量方程耦合非常密切时，SIMPLEC 和 SIMPLER 算法的效果可能更好些。

另外还有瞬态问题的算法、同位网格的算法等，限于篇幅，请读者查阅相关书籍。

第 2 章 Fluent 基本介绍

Fluent 是目前国际上比较流行的商用 CFD 软件包，用来模拟从不可压缩到高度可压缩范围内的复杂流动。由于采用了多种求解方法和多重网格加速收敛技术，因而 Fluent 能达到最佳的收敛速度和求解精度。灵活的非结构化网格和基于解的自适应网格技术及成熟的物理模型，以及强大的前后处理功能，使 Fluent 在转折与湍流、传热与相变、化学反应与燃烧、多相流、旋转机械、动/变形网格、噪声、材料加工、燃料电池、航空航天、汽车设计、石油天然气、涡轮机设计、环境工程、安全工程等方面有着广泛应用。

2.1 Fluent 概述

2.1.1 Fluent 软件组成

Fluent 软件的基本程序结构如图 2-1 所示，一般包括以下几个部分：

(1) Fluent，进行模拟计算的求解器。

(2) prePDF，用于模拟 PDF 燃烧过程。

(3) Gambit，建立几何模型及网格生成。

(4) TGrid，用于从现有的边界网格生成体网格。

(5) Filters(Translators)，转换其他程序生成的网格，用于 Fluent 计算。

可以接口的程序包括 Ansys、I-DEAS、Nastran、Patran 等。

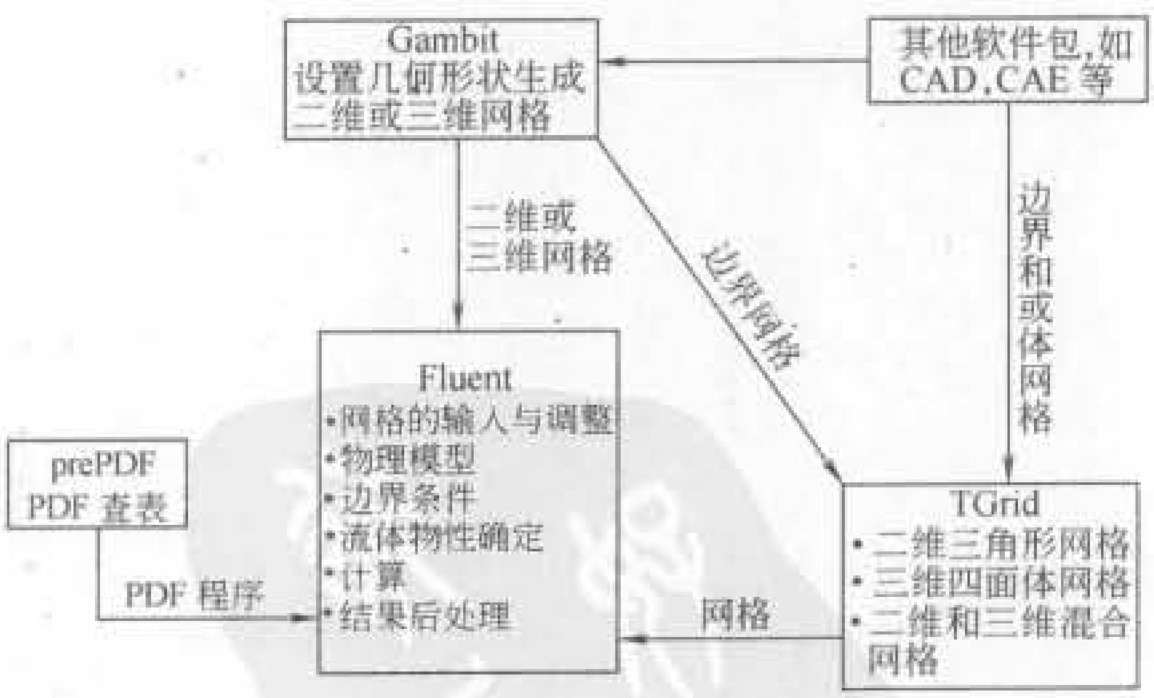


图 2-1 Fluent 程序结构图

2.1.2 Fluent 的文件类型

使用 Fluent 时，需要输入和输出几种类型的文件，其中读入的文件包括 grid、cas、data、profile、scheme 以及 journal 文件，输出文件包括 case、data、profile、journal 以及 transcript 的文件。Fluent 还可以保存面板的布局以及图形窗口的硬拷贝，使用各种可视化以及后处

理工具可以输出数据。其主要文件类型如表 2-1 所示。

表 2-1 Fluent 中的主要文件类型

文件名称	扩 展 名	内 容
Grid (网格文件)	msh	包含所有节点的坐标及节点间的连接信息，不含边界条件、流动参数及求解参数
Case(案例文件)	cas	包括网格、边界条件、求解参数、流动参数、物性参数、用户界面和图形环境的信息
Data(数据文件)	dat	包含每个网格单元的流场值及收敛的历史记录(残差值)
Profile(边界信息文件)	用户指定	指定边界区域上的流动条件，以及动边界的运动等条件
Journal(日志文件)	用户指定	记录用户输入过的各种命令
Transcript(副本文件)	用户指定	记录全部输入及输出信息
Hardcopy(硬拷贝文件)	取决于输出格式	将图形窗口中的内容硬拷贝输出为 TIFF、PICT 和 BMP 等格式的文件
Export(输出文件)	取决于输出格式	将数据输出到 AVS、DataExplorer、Ensight、FAST、FIELDVIEW、I-DEAS、NASTRAN、PATRAN 及 Tecplot 等第三方 CAD/CAE 软件
Scheme(源文件)	scm	定制 Fluent 的界面、控制 Fluent 的运行。Scheme 语言是 LISP 编程语言的一个分支，用 Scheme 语言编写的源程序文件称为 Scheme 文件
Interpolation(转接文件)	用户指定	不同网格求解时的转接文件，Fluent 允许用户用两种网格方案求解同一问题，中间通过 Interpolation 文件交换数据。具体过程是，先用第一种网格求解，然后将求解的结果保存为 Interpolation 文件，再接着用第二套网格方案从该起点处继续往下计算
Fluent(配置文件)	Fluent	包含用 Scheme 语言写成的语句，用于对 FIUENT 进行定制和控制。Fluent 启动时，要寻找该文件，若找到它，就加载它
cxlayout(界面布局文件)	cxlayout	保存当前对话框及图形窗口的布局

2.1.3 Fluent 的特点

(1) Fluent 软件采用基于完全非结构化网格的有限体积法，而且具有基于网格节点和网格单元的梯度算法，网格类型较多，如图 2-2 所示。

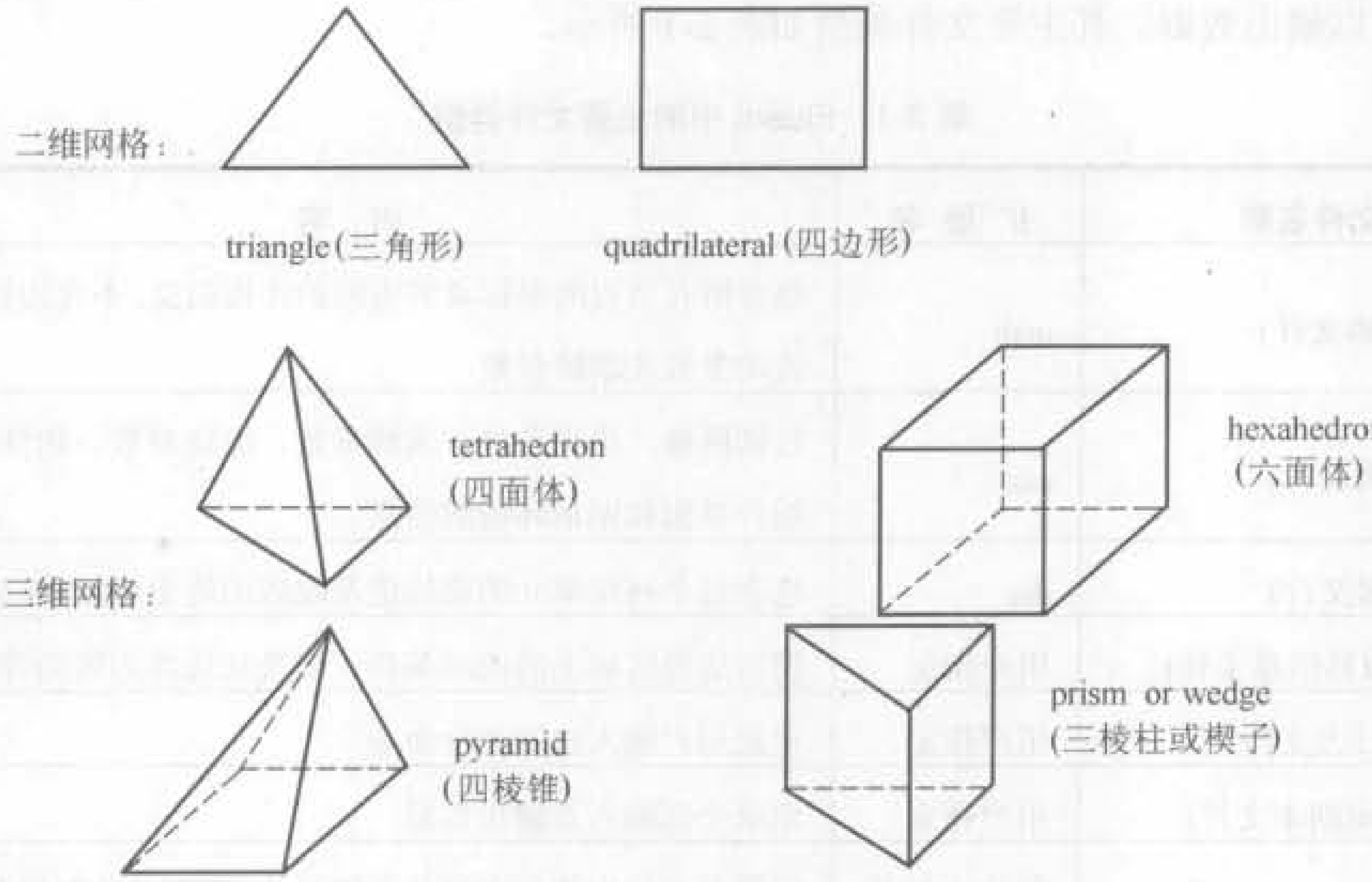


图 2-2 Fluent 的网格(控制体积)形状

(2) 定常/非定常流动模拟，而且新增快速非定常模拟功能。

(3) Fluent 软件中的动/变形网格技术主要解决边界运动的问题，用户只需指定初始网格和运动壁面的边界条件，余下的网格变化完全由解算器自动生成。网格变形方式有三种：弹簧压缩式、动态铺层式以及局部网格重生式。其局部网格重生式是 Fluent 所独有的，而且用途广泛，可用于非结构网格、变形较大问题以及物体运动规律事先不知道而完全由流动所产生的力所决定的问题。

(4) Fluent 软件具有强大的网格支持能力，支持界面不连续的网格、混合网格、动/变形网格以及滑动网格等。值得强调的是，Fluent 软件还拥有多种基于解的网格的自适应、动态自适应技术以及动网格与网格动态自适应相结合的技术。

(5) Fluent 软件包含三种算法：非耦合隐式算法、耦合显式算法、耦合隐式算法，所用算法是商用软件中最多的。

(6) Fluent 软件包含丰富而先进的物理模型，使得用户能够精确地模拟无粘流、层流和湍流。湍流模型包含 Spalart-Allmaras 模型、k-ω 模型组、k-ε 模型组、雷诺应力模型(RSM)组、大涡模拟模型(LES)组以及最新的分离涡模拟(DES)和 V2F 模型等。另外用户还可以定制或添加自己的湍流模型。

(7) 适用于牛顿流体和非牛顿流体。

(8) 含有强制/自然/混合对流的热传导，固体/流体的热传导、辐射。

(9) 化学组分的混合/反应。

(10) 包含自由表面流模型、欧拉多相流模型、混合多相流模型、颗粒相模型，空穴两相流模型以及湿蒸汽模型。

(11) 熔化凝固，蒸发/冷凝相变模型。

(12) 离散相的拉格朗日跟踪计算。

(13) 非均质渗透性、惯性阻抗、固体热传导，多孔介质模型(考虑多孔介质压力突变)。

(14) 风扇、散热器、以热交换器为对象的集中参数模型。

(15) 惯性或非惯性坐标系、复数基准坐标系及滑移网格。

(16) 动静翼相互作用模型化后的接续界面。

(17) 基于精细流场解算的预测流体噪声的声学模型。

(18) 质量、动量、热、化学组分的体积源项。

(19) 丰富的物性参数的数据库。

(20) 磁流体模块主要模拟电磁场和导电流体之间的相互作用问题。

(21) 连续纤维模块主要模拟纤维和气体流动之间的动量、质量以及热的交换问题。

(22) 高效率的并行计算功能，提供多种自动/手动分区算法；内置 MPI 并行机制可大幅度提高并行效率。另外，Fluent 特有动态负载平衡功能，确保全局高效并行计算。

(23) Fluent 软件提供了友好的用户界面，并为用户提供了二次开发接口(UDF)。

(24) Fluent 软件采用 C/C++语言编写，从而大大提高了对计算机内存的利用率。

2.2　Fluent 的操作界面

Fluent 控制台是控制程序执行的主窗口。用户和控制台之间有两种交流方式：图形界面(GUI)、文本界面(TUI)。控制包括菜单按钮的图形界面和终端仿真程序。

2.2.1　图形用户界面

Fluent 图形用户界面由下拉菜单、控制面板、对话框以及图形窗口组成。图 2-3 就是典型的 Fluent 6.2 界面。在 UNIX 系统中，GUI (包括颜色和字体)可以自定义以适合操作系统的环境。

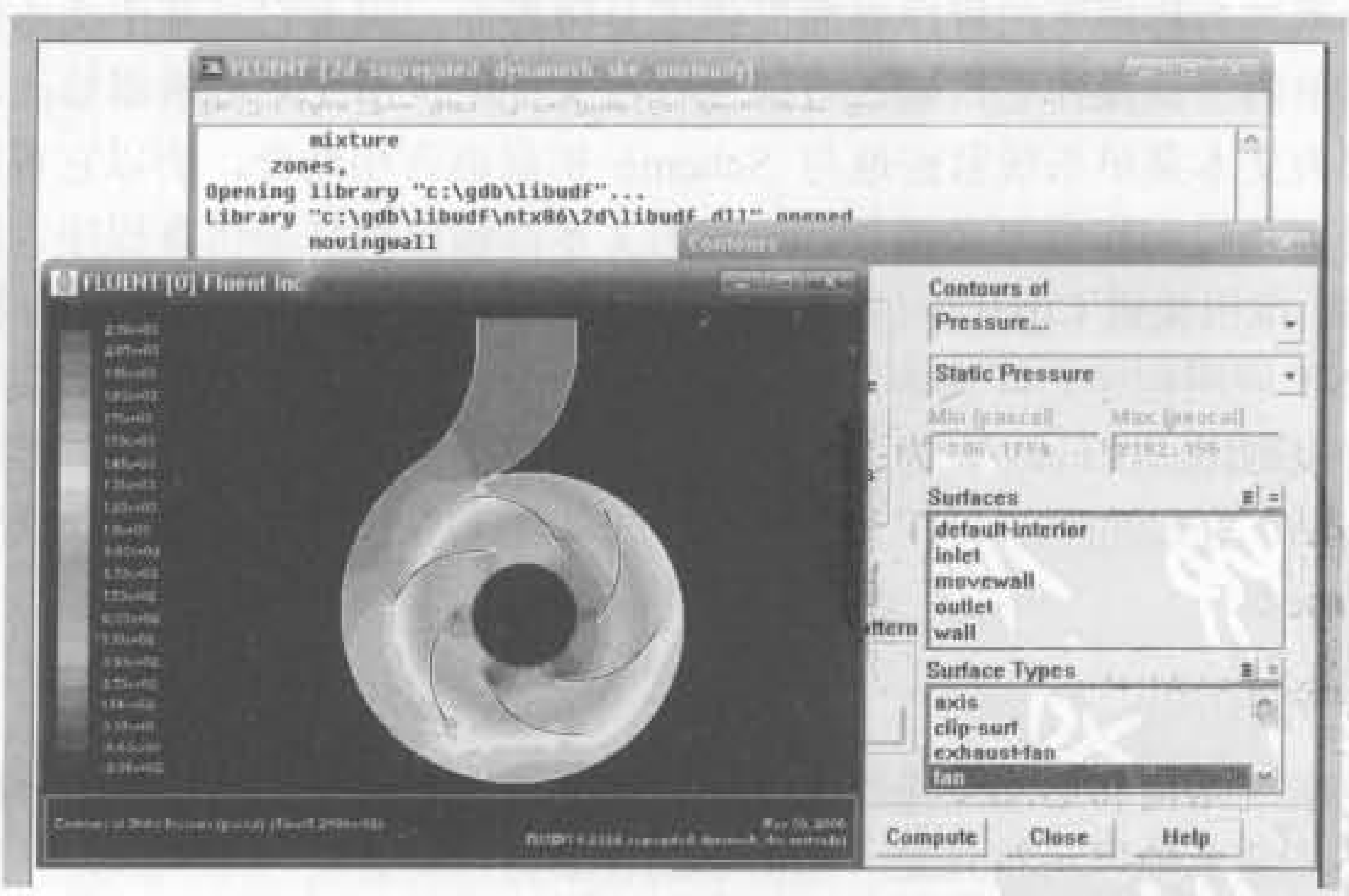


图 2-3　Fluent 界面

Fluent 菜单按钮提供了控制台大部分命令，下拉菜单使用方法和 Windows 的一样，快捷方式也采用按 Alt 键和下划线字母键选中，按 Esc 键退出。有些下拉菜单有快捷键，在

相应的菜单后面会提示快捷键的字母。

Fluent 中的对话框用于完成简单的输入/输出任务，比如说警告、错误和询问。对话框是临时窗口，出现时要对其进行响应。如信息提示框告诉我们需要知道的信息，单击 OK 按钮就关闭了；又如警告对话框用于警告某些潜在问题，并询问是否继续当前操作。

面板用于处理复杂的输入任务。和对话框相似，面板也是一个独立的窗口，但是使用面板更像是填充一个表格。每一个面板都是独一无二的，而且使用各种类型的输入控制组成表格。

显示选项面板可以控制图形显示的属性也可以打开另一个显示窗口。鼠标按钮面板可以用于设定鼠标在图形显示窗口单击时所执行的操作。当为图形显示处理数据时要取消显示操作可以按快捷键 Ctrl+C，已经开始画图的话就无法取消操作了。

在 Windows NT 环境下的 Fluent，单击图形窗口的左上角便可以显示图形窗口系统菜单，该菜单包括常用系统命令，如 move、size 和 close。连同系统命令一起，Fluent 为支持打印机和剪贴板增加了三条命令。

(1) 复制到剪贴板：将当前图形复制到 Windows 的剪贴板。可以用页面设置面板改变复制的属性。图形窗口的大小影响了图形中所使用的字体的大小。

(2) 打印：将当前图形复制到打印机。可以用页面设置面板改变打印的属性。

(3) 页面设置：显示页面设置面板。

同时还有页面设置面板功能：在图形显示窗口中选择 System→Page Setup 菜单命令，利用其中的命令可对所需输出图形格式等参数进行调整。

2.2.2 文本用户界面(TUI)及 Scheme 表达式

1. 概述

文本菜单系统为程序下的程序界面提供了分级界面。因为它是基于文本的，所以可以用标准基于菜单的工具操作它，输入可以保存在文件中，用文本编辑器修改，并可以将执行的读入。因为文本菜单系统紧密地与 Scheme 扩展语言相结合，所以它可以很容易地形成程序来提供复杂控制和自定义函数。所有的文本都输出到终端仿真程序，所有的输入都从最底行开始。按快捷键 Ctrl+C 可以暂停正在计算的程序。它也支持控制台和其他 X 窗口或 Windows NT 应用程序之间文本的复制和粘贴。文本可以在控制台窗口复制和粘贴。在 UNIX 系统中复制和粘贴的方法为：①用鼠标左键选中要复制的文本；②到新窗口单击鼠标中键便可粘贴。在 Windows NT 系统中复制文本到剪贴板的方法为：①选中文本；②按快捷键 Ctrl+Insert。

2. TUI 命令系统结构

文本菜单系统结构和 UNIX 操作系统(或 DOS)的目录树很相似。当第一次进入 Fluent，是在根菜单下，菜单的提示符只是一个简单的补字符“>”。

要生成子菜单和命令的列表只需按 Enter 键：

```
>
adapt/                  grid/                   surface/
display/                plot/                   view/
```

```
define/          report/       exit
file/            solve/
```

TUI 中的子菜单的名字都以"/"结尾，以区别于菜单命令。要执行一个命令，输入命令名或该命令的简写就可以。与之相似，进入子菜单，只需输入菜单名字或其简写就可以，提示符也会相应改变为当前菜单的名字。

```
> display
/display> set
/display/set>
```

要回到上一级菜单只需在命令提示中输入 q 或者 quit。

```
/display/set> q
/display>
```

可以输入菜单全路径名直接进入到另一菜单。

```
/display> /file
/display//file>
```

上面控制直接从/display 转到/file 而不结束根菜单，因此，当从/file 菜单退出时，控制会直接退回到 display，如

```
/display//file> q
/display>
```

而且，如果直接执行一个命令而不结束路径上的任何菜单，控制会仍然回到调用命令时的菜单。

```
/display> /file start-journal jrnl
Input journal opened on file "jrnl".
/display>
```

3. 命令缩写

在 TUI 模式下，可以输入匹配该命令的缩写。匹配命令的规则如下：命令由连字符分隔的短语组成，该命令与短语的初始序列匹配，连字符的匹配是可选的，短语和它的字符串的初始序列匹配，通过输入那个字符串来匹配。

如果一个缩写匹配多于一个命令，那么具有最大匹配字符数的命令将被选择。如果不止一个命令有相同的匹配短语，那么第一个出现在菜单中的命令将被选择。

例如，下面的每一个都匹配命令 set-ambient-color：set-ambient-color、s-a-c、sac 和 sa。当缩写命令时，通常缩写会匹配不止一条的命令。在这种情况下，第一个命令将会被选择。有时候会有不正常的情况，比如说 lint 不匹配 lighting-interpolation，因为 li 匹配 lights-on，但是 nt 并不匹配 interpolation，这一问题可以通过选择不同的缩写来解决，如 liin 或者 l-int。

4. 命令别名

在 TUI 中可以定义命令的别名。在 UNIX csh shell 中，别名比命令执行的优先级要高。下面的别名是在 Cortex 中预定义的：error、pwd、chdir、ls 以及 alias。

error：显示最近 Scheme 错误中断中无效 Scheme 对象。

pwd：打印工作目录，在这个工作目录中所有的文件操作都可以进行。

chdir：改变工作目录。

ls：列出工作目录的文件。

alias：显示当前别名的符号列表。

5. Scheme 赋值

如果在菜单提示行中输入“(”，那么所有的插入语和所有的字符串加上“)”都会传送到被赋值的 Scheme 中，而且赋值的表达式显示如下：

```
> (define a 1)  a  > (+ a 2 3 4)  10
```

所有的响应(除了文件名)在被使用之前都被 Scheme 解释程序赋值了，因此可以输入任何一个有效的 Scheme 表达式来响应提示。例如，输入一个单位矢量，某一分量为 1/3。

```
/foo> set-xy  x-component [1.0] (/ 1 3)  y-component [0.0] (sqrt (/ 8 9))
```

或者你可以输入一个有效函数，计算单位矢量的另一个分量。

```
> (define (unit-y x) (sqrt (- 1.0 (* x x))))  unit-y  /foo> set-xy
x-component [1.0] (/ 1 3)  y-component [0.0] (unit-y (/ 1 3))
```

6. 文本提示系统

命令需要各种变量：数、文件名、Yes/No 响应、字符串和列表。这些输入的统一界面是一个文本提示系统，提示包括提示字符串以及相应的用方括号括起来的选项或者用方括号括起来的默认值。如 filled grids? [no]，shrink-factor [0.1]，line-weight [1]，title [""]获取提示的默认值只需要按 Enter 键或者输入逗号。

注意： 逗号不是一个分隔符，它是默认值的分隔标志：“1,2” 表示 3 个值，“1” 是第一个提示值，第二个提示值为默认，“2” 为第三个提示值。在任何提示中输入 “a” 会显示一个简短的帮助信息。要中断一个提示序列只需按快捷键 Ctrl+C 即可。

7. 数

一般大多数的提示类型是数，既可是整数也可是实数。举例来说，16、-2.4、.9e5 和 +1e-5 是有效地输入。整数也可以是二进制、八进制和十六进制的格式。如：十进制数 31 可以输入为 31、#b11111、#o37 或者#x1f。在 Scheme 中，整数是实数的子集，所以不需要加上小数点表明哪一个数是实数，2 也是实数 2.0。如果在整数提示符中输入实数，那么小数部分会被省掉，如 1.9 就变成 1 了。

8. 布尔运算符

有些提示需要 yes 或 no 的响应。yes 或 y 表示同意，no 或者 n 表示不同意。yes/no 提示通常用于证实某些潜在的危险操作，如覆盖文件、不保存文件就退出、数据、网格等是否进行。有些提示符需要真正的布尔值(真或假)，其输入分别为#t 和#f。

9. 字符串

字符串的输入需要双引号括起，如“red”。绘制标题或者绘制图例就是字符串的一个例子。字符串可以包括任何字符，包括空格和标点。

10. 符号

符号的输入不需要加引号。区域名\表面名以及材料名就是符号的例子。符号必须以字母开始，不能包括任何空格或逗号。

11. 文件名

文件名只是字符串的一种，为方便起见，文件名不需要用双引号括起来。如果有些例外——文件名中有空格，那么文件名必须用双引号括起来。

这样“方便”结果使得文件名提示无响应值。例如：> (define fn "valve.ps") fn > hc fn 会结束 fn，文件名的硬拷贝，而不是 valve.ps。因为文件名提示无响应值，fn 没有机会求“valve.ps”的值，对于大多数其他的提示也是一样。

12. 列表

Fluent 中有些函数需要目标的列表，如数、字符串、布尔运算值等。Scheme 对象的列表是一个简单的由空白列表“'()”结束的对象序列。每次列表提示一个单元，最后一个是空列表。这一结束列表组成了提示列表的末尾，既可能是空也可能包含任何值。为方便起见，空列表中可以输入“()”也可以输入标准格式“'()”。通常地，列表提示默认保存先前声明的列表。要修改列表，覆盖所需单元并用空列表结束进程。例如：element(1) [()] 1，element(2) [()] 10，element(3) [()] 100，element(4) [()]相应地创建三个数 1、10 和 100 的列表；element(1) [1],. element(2) [10]，element(3) [100]，element(4) [()] 1000，element(5) [()]增加第四个单元。然后 element(1) [1]，element(2) [10]，element(3) [100] () 只有 1 和 10 在列表中。随后输入 element(1) [1] ,,'(11 12 13) 创建一个五元素列表：1、10、11、12 和 13。最后一个空列表移走所有的单元 element(1) [1] ()。

13. 默认值绑定

任何提示的默认值被限制为 Scheme 符号“_”(下划线)，以便于默认值可以形成 Scheme 表达式的一部分。例如，如果想将默认值减去，可以输入 shrink-factor [0.8] (/ _ 3)。

14. 中断

代码的执行可以用 Ctrl+C 组合键停止，这时，目前的操作停止在下一个可恢复的位置。

15. 系统命令

如果在 UNIX 操作系统中运行 Fluent，可以用字符! (bang)来执行系统命令。在 UNIX 基础的操作系统下可以执行系统命令。以!开始的所有字符串一直到下一行开始都会在 subshell 中执行。与这些系统命令有关的任何进一步的输入必须被输入到启动程序的窗口中，而且任何的输出也是在这个窗口中。(注意：如果远程启动 Fluent，这些输入和输出必须是在启动 shell(cortex)的窗口中。

```
> !rm junk.* > !vi script.rp
```

别名 ls 和 pwd 在工作目录中调用 UNIX ls 和 pwd 命令，别名 chdir 改变程序目前的工作目录。

!ls 和!pwd 将会在 shell 启动的目录中执行 UNIX 命令。屏幕输出会在启动 Fluent 的窗口中，除非使用远程启动，在远程启动中会在启动 shell 的窗口中输出。(注意：!chdir 或者!cd 在 subshell 中执行，所以它不会改变 Fluent 或者 Cortex 的工作目录，因此它并不是很有用)。不带任何声明的输入 chdir 会将你移到控制台的父目录。

2.2.3 图形控制及鼠标使用

(1) 图形控制主要是通过菜单 display 和 plot 进行显示内容选择和显示属性设置。例如，要显示计算结果的各类云图(或等值线)，可以通过 display/contours 进行压力、速度等物理量的云图(或等值线)显示。通过 display/options 进行图形窗口的显示属性的修改。单击图形窗口左上角(或右击图形窗口的上框)，弹出一对话框，便于用户复制或打印图形，及改变图形窗口的背景或网格颜色。在命令行直接输入 display/set/color 命令，然后输入要改变的颜色的对象名称(直接按 Enter 键显示所有的对象名称)也可以达到修改图形窗口的背景或网格颜色的目的。

(2) 鼠标在图形窗口的功能为：按住左键拖动，移动图形；按住中键拖动，缩放图形；按住右键拖动，执行用户预定义的操作。鼠标按键的调整是通过 display/mouse buttons 命令完成的。

2.3 Fluent 简单操作实例

2.3.1 Fluent 计算流程

(1) 创建网格。

(2) 运行合适的解算器：2D、3D、2DDP、3DDP。

(3) 输入网格。

(4) 检查网格。

(5) 选择解的格式。

(6) 选择需要解的基本方程：层流还是湍流(无粘)、化学组分还是化学反应、热传导模型等。

(7) 确定所需要的附加模型：风扇、热交换、多孔介质等。

(8) 指定材料物理性质。

(9) 指定边界条件。

(10) 调节解的控制参数。

(11) 初始化流场。

(12) 计算解。

(13) 检查结果。

(14) 保存结果。

(15) 必要的话，细化网格，改变数值和物理模型。

2.3.2　简单流动与传热的计算

下面以射流冷热水混合器内部流场温度场模拟为例说明 Fluent 的基本操作过程，射流冷热水混合器结构如图 2-4 所示。启动 Fluent 的 2D 求解器后顺序进行如下操作。



图 2-4　射流冷热水混合器结构

1. 读入河流网格文件 jet-reactor.msh(见网站下载资源中的 jet-reactor.msh)

操作：选择 File→Read→Case 命令。

找到网格文件，单击 OK 按钮打开，其窗口反馈信息如图 2-5 所示，其中包括节点、网格数目等信息。



图 2-5　读入网格文件的信息反馈

2. 检查网格

操作：选择 Grid→Check 命令。

其信息反馈窗口如图 2-6 所示。

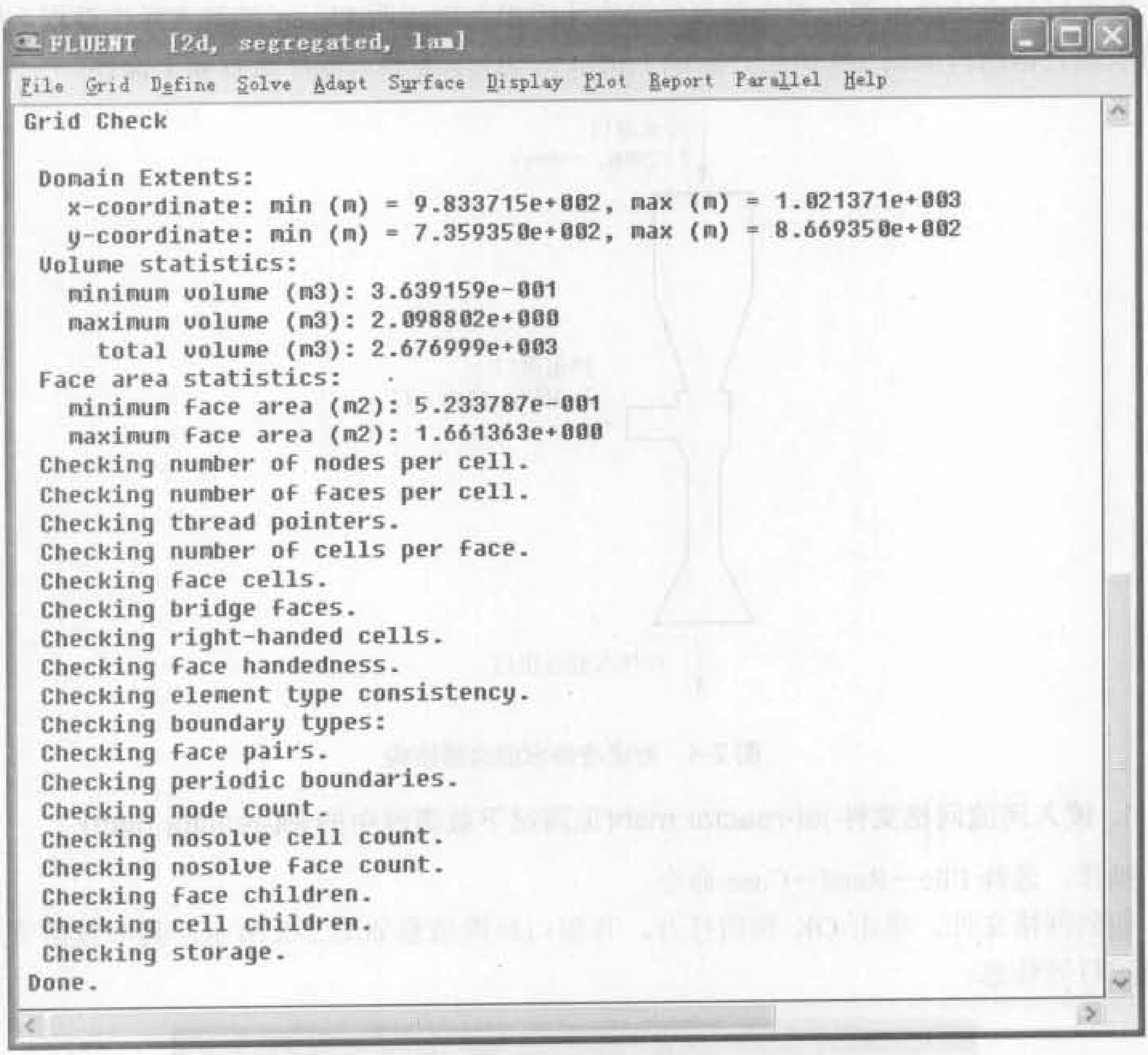


图 2-6 检查网络的信息反馈

网格检查时主要反馈了以下信息：①列出了 x、y 的最小和最大值；②列出了网格的其他特性，如单元的最大体积和最小体积、最大面积和最小面积等；③报告网格的任何错误。网格检查时要特别注意最小体积的数值，要确保其不能为负。

3. 平滑(和交换)网格

操作：选择 Grid→Smooth/Swap 命令。

打开 Smooth/Swap Grid(平滑与交换网格)对话框，如图 2-7 所示。单击 Smooth 按钮，单击 Swap 按钮，重复上述操作，直到 Fluent 报告没有需要交换的面为止，如图 2-8 所示，这一功能对三角形网格尤为重要。

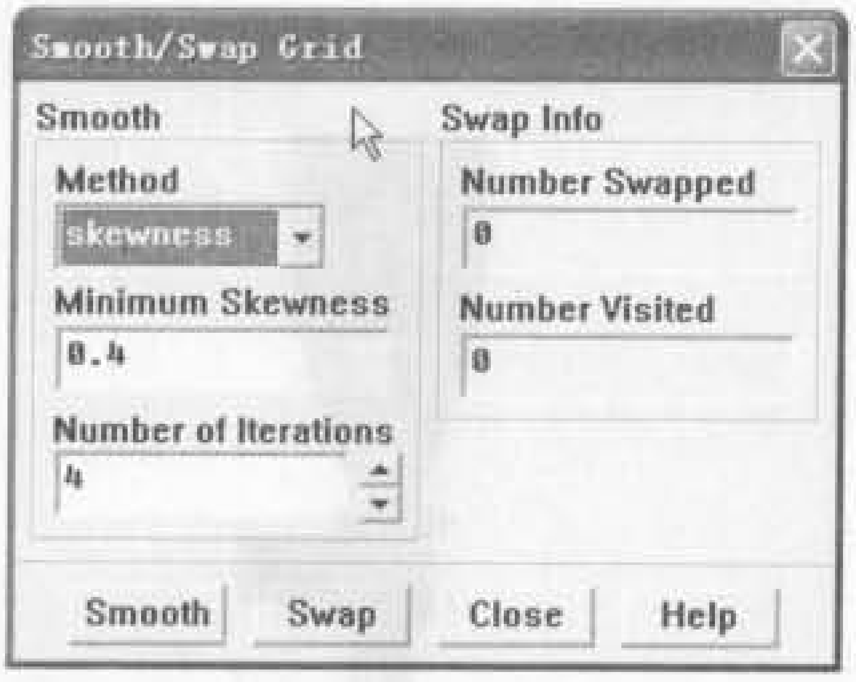


图 2-7　Smooth/Swap Grid 对话框

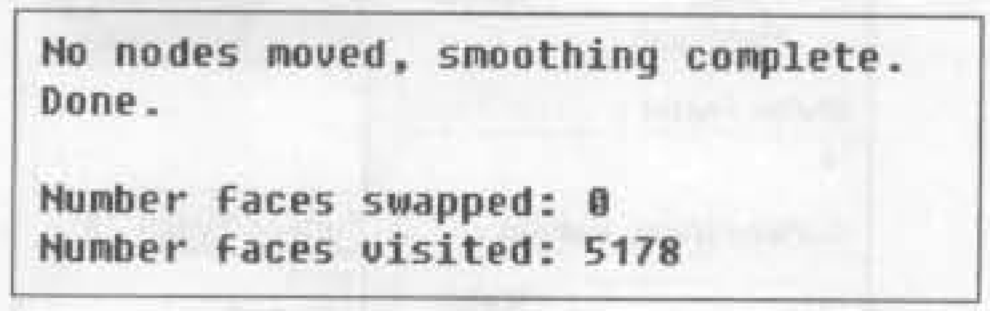


图 2-8　平滑与交换网格信息反馈

4. 确定长度的单位

操作：选择 Grid→Scale 命令。

打开 Scale Grid(标定网格)对话框，如图 2-9 所示。在 Unit Conversion(单位转换)选项组中的 Grid Was Created In 下拉列表框中选择 mm；单击 Change Length Units 按钮，此时，在 Domain Extents 选项组中给出了区域的范围和度量的单位；单击 Scale 按钮，完成单位转换。

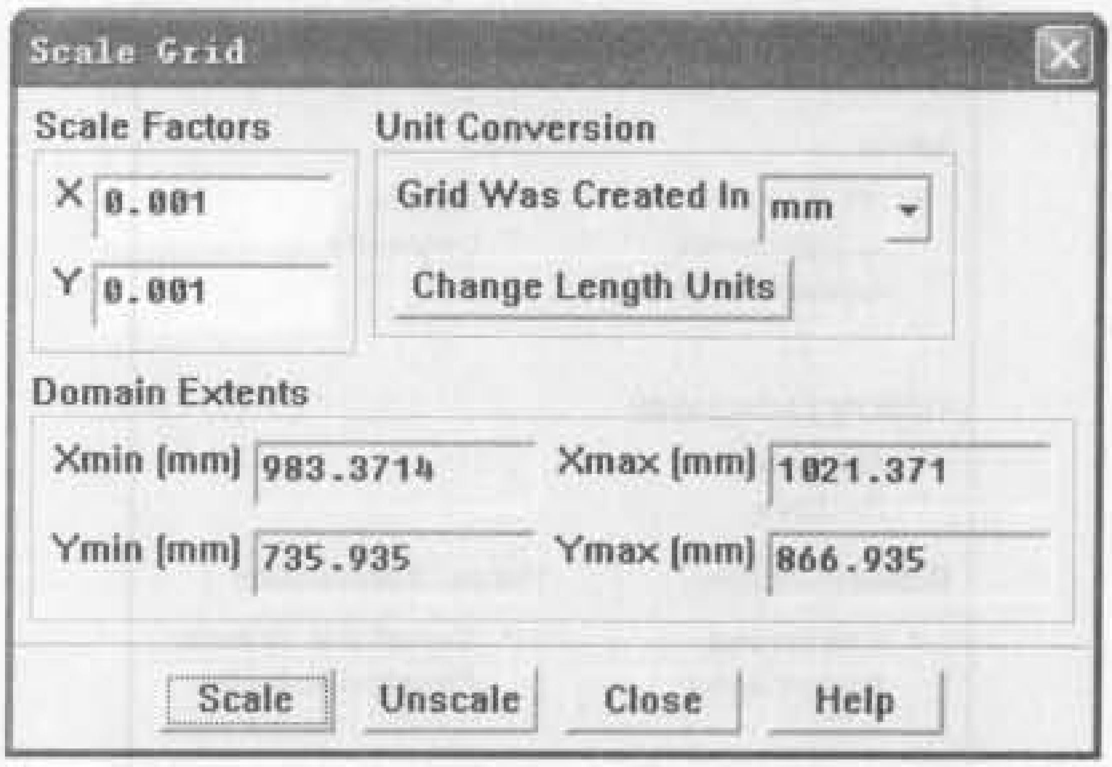


图 2-9　Scale Grid 对话框

在 Fluent 中，除了长度单位外，其他单位均采用 SI 制。一般不需要改动，若要对单位进行改动，应启动 Set Units(设置单位)对话框进行修改，命令在 Define→Units 中。

5. 显示网格

操作：选择 Display→Grid 命令。

打开 Grid Display(网格显示)对话框，如图 2-10 所示。在 Surfaces 选项组中选择所有的表面；单击 Display 按钮，显示本例的网格图，如图 2-11 所示。

在图形窗口，右击边界线，在信息反馈窗口内将显示此边界的类型等信息。也可用此方法检查内部任意节点和网格线的信息，便于边界条件的设置。

6. 设置求解器

操作：选择 Define→Models→Solver 命令。

打开 Solver 对话框，如图 2-12 所示。

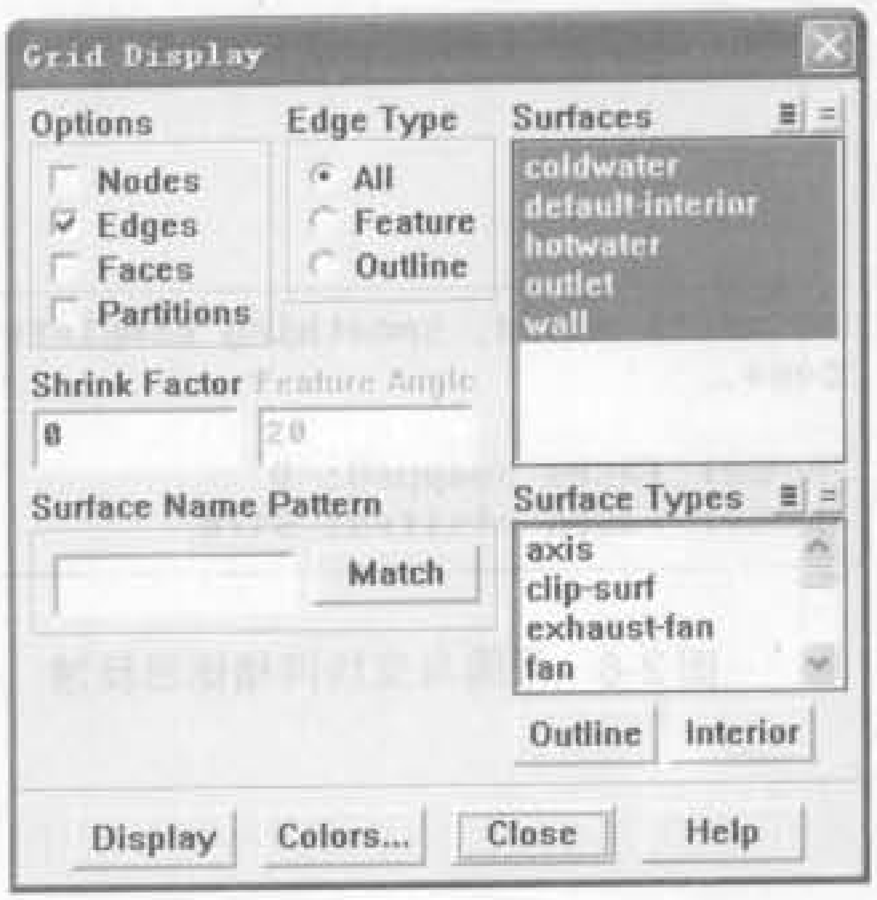


图 2-10 Grid Display 对话框

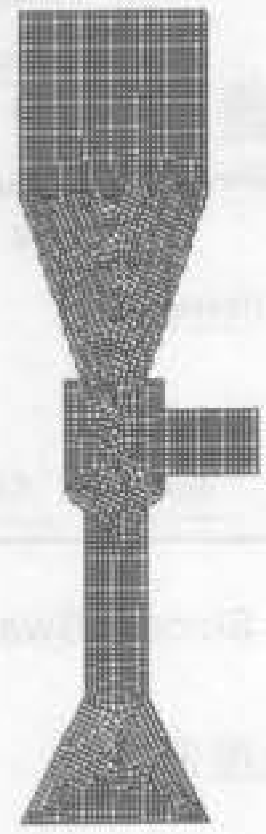


图 2-11 网格图

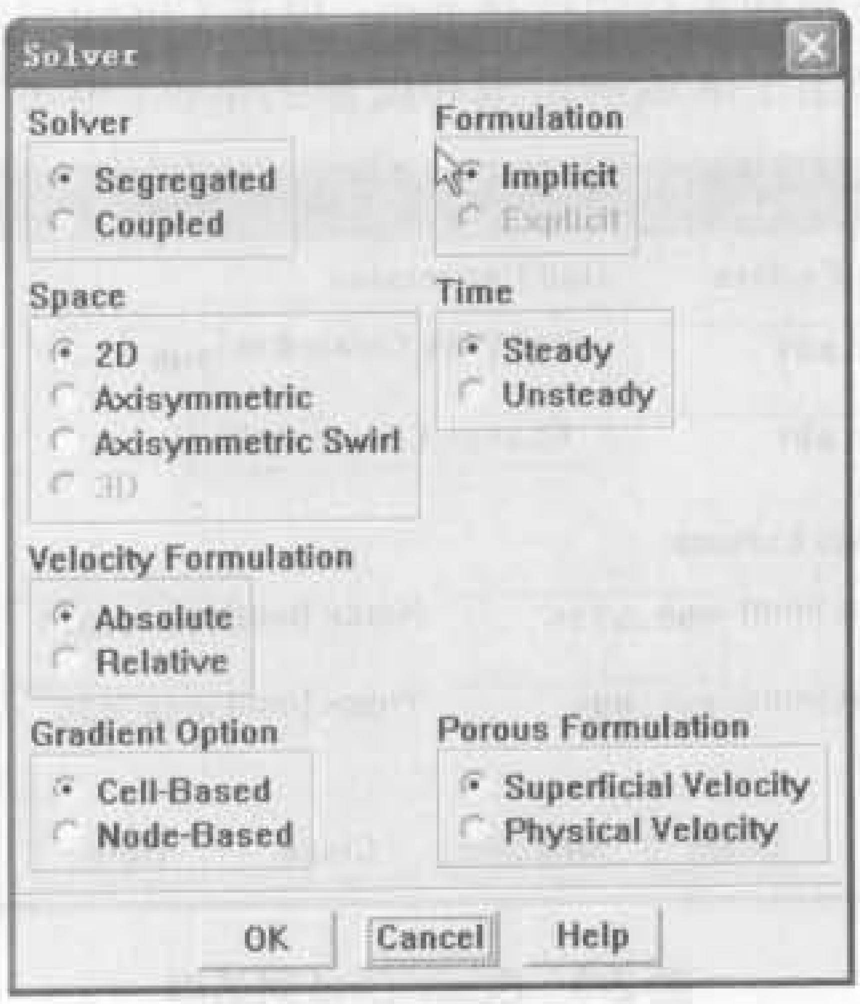


图 2-12 Solver 对话框

图 2-12 中：①Solver 选项组用于选择求解器：Segregated 的含义是非耦合求解法；Coupled 的含义是耦合求解法。②Formulation 选项组用于选择算法：Implicit 的含义是隐式算法；Explicit 的含义是显式算法。③Space 选项组用于选择空间属性：2D 的含义是二维空间；Axisymmetric 的含义是轴对称空间；Axisymmetric Swirl 的含义是轴对称旋转空间；3D 的含义是三维空间。④Time 选项组用于选择时间属性：Steady 的含义是定常流动；Unsteady 的含义是非定常流动。⑤Velocity Formulation 选项组用于选择速度属性：Absolute 的含义是绝对速度；Relative 的含义是相对速度。⑥Gradient Option 选项组用于选择梯度选项：Cell-Based 的含义是基于单元梯度；Node-Based 的含义是基于节点梯度。⑦Porous Formulation 选项组用于选择多孔算法：Superficial Velocity 的含义是表面速度算法；Physical Velocity 的含义是物理速度算法。这里保持默认设置。

7. 设置标准 *k*-湍流模型

操作：选择 Define→Models→Viscous 命令。

打开 Viscous Model(粘度模型)对话框的选择界面，如图 2-13 所示，Inviscid 表示无粘(理想)流体；Laminar 表示层流模型；另外 4 个为常见的湍流模型。选中 k-epsilon 单选按钮，则打开 Viscous Model 对话框的设置界面，如图 2-14 所示，保持默认设置。



图 2-13　Viscous Model 对话框的选择界面

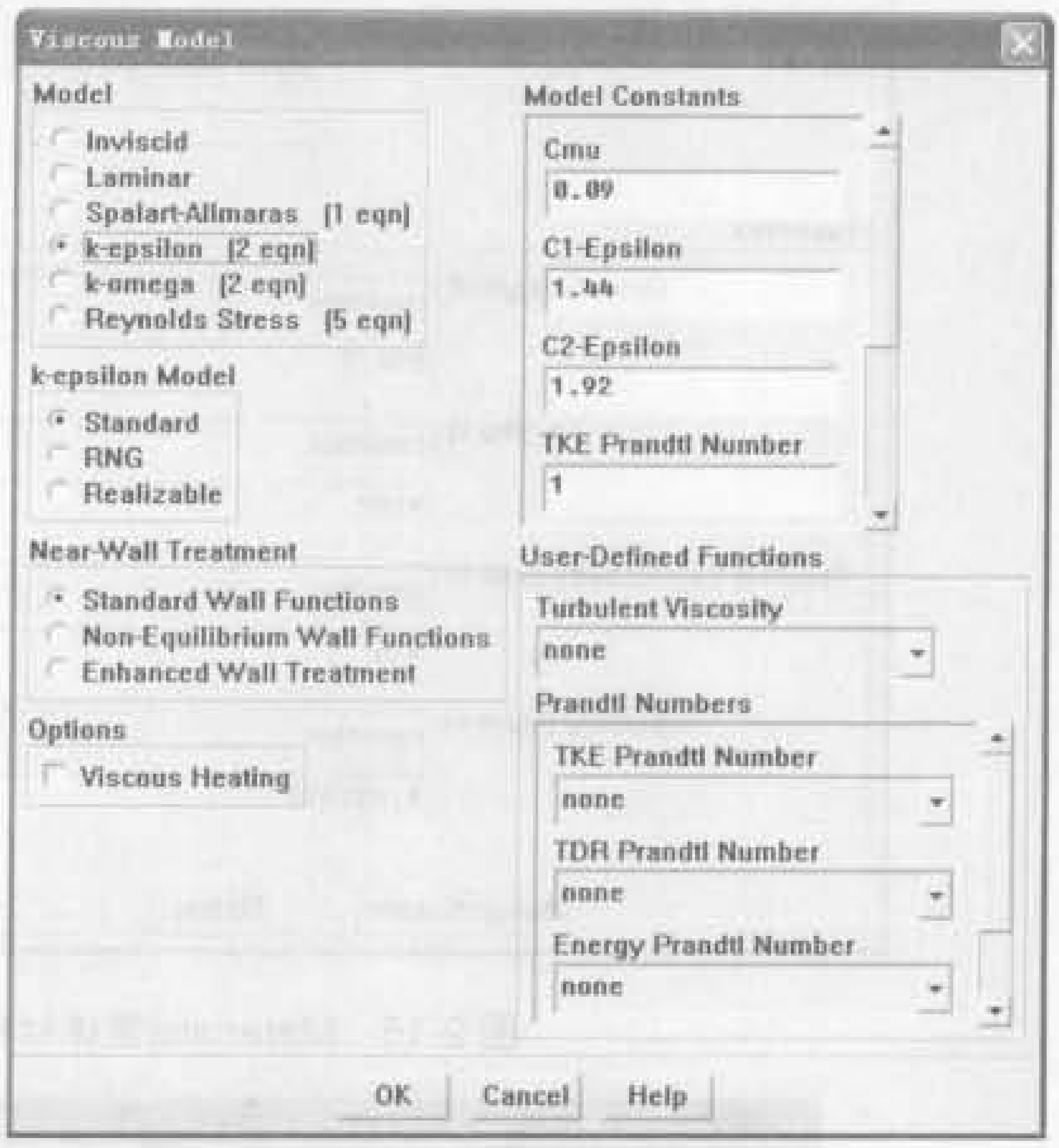


图 2-14　Viscous Model 对话框的设置界面

8. 选择能量方程

操作：选择 Define→Models→Energy 命令。

打开如图 2-15 所示的 Energy 对话框，选中 Energy Equation(能量方程)复选框。

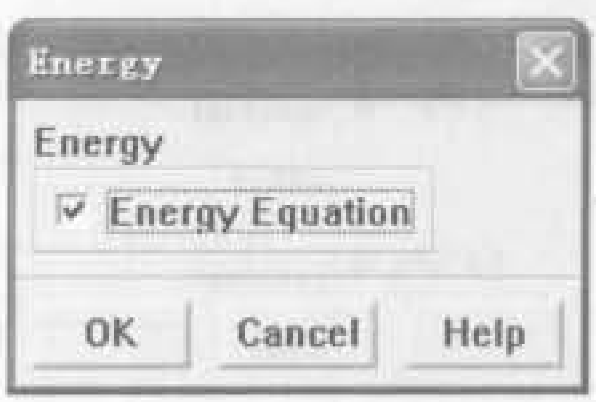


图 2-15　Energy 对话框

9. 设置流体的物理属性

从 Fluent 材料库中选择 water 流体。

操作：选择 Define→Materials 命令。

打开 Materials(流体材料设置)对话框，如图 2-16 所示，单击 Fluent Database 按钮，打开 Fluent Database Materials(Fluent 材料库选择)对话框；选择 water-liquid(h2o<l>)流体，如

图 2-17 所示，保持默认设置，单击 Copy 按钮，单击 Close 按钮关闭 Fluent Database Materials 对话框。在 Materials 对话框中单击 Change/Create 按钮，单击 Close 按钮关闭 Materials 对话框。

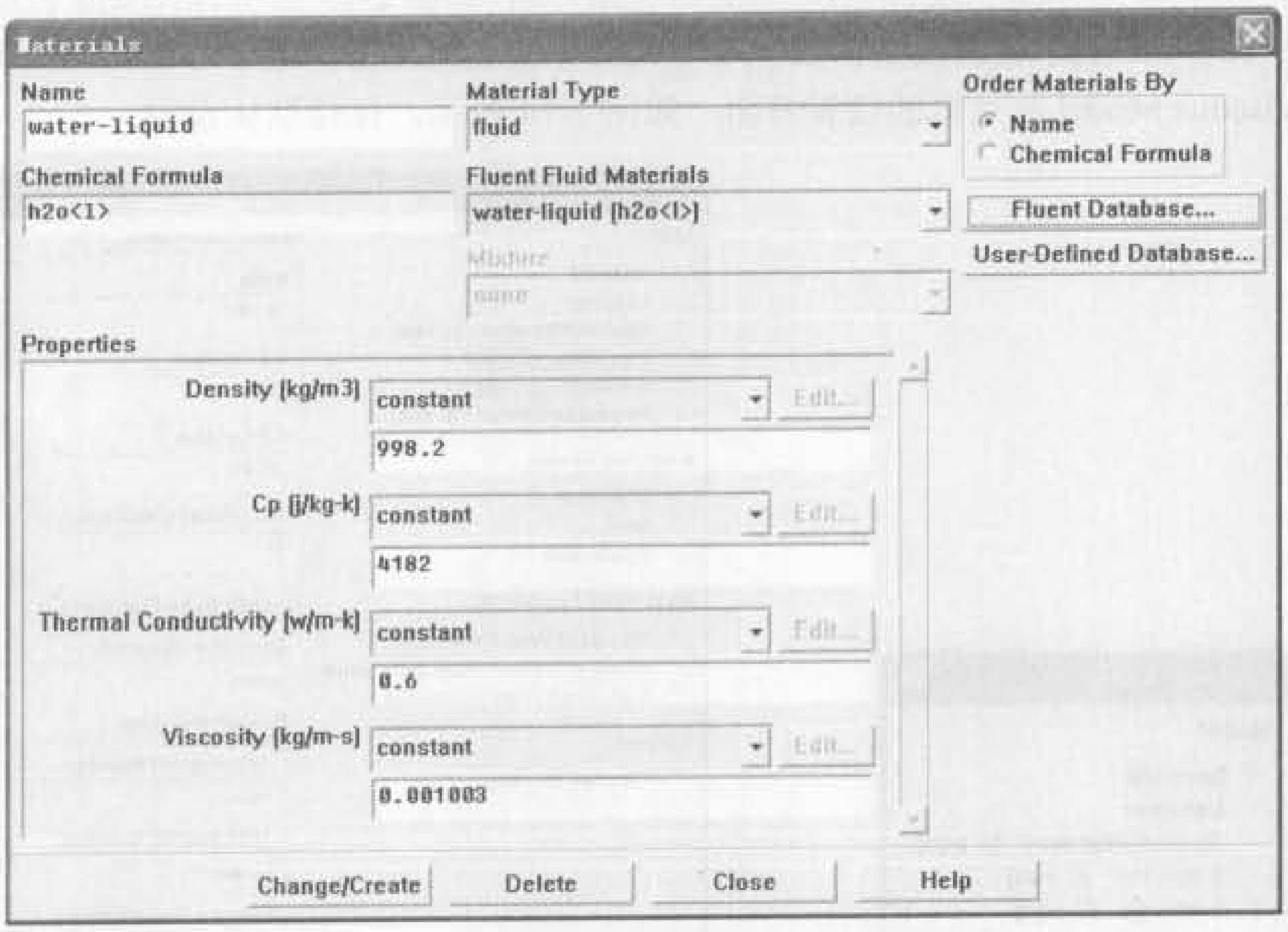


图 2-16　Materials(流体材料参数设置)对话框

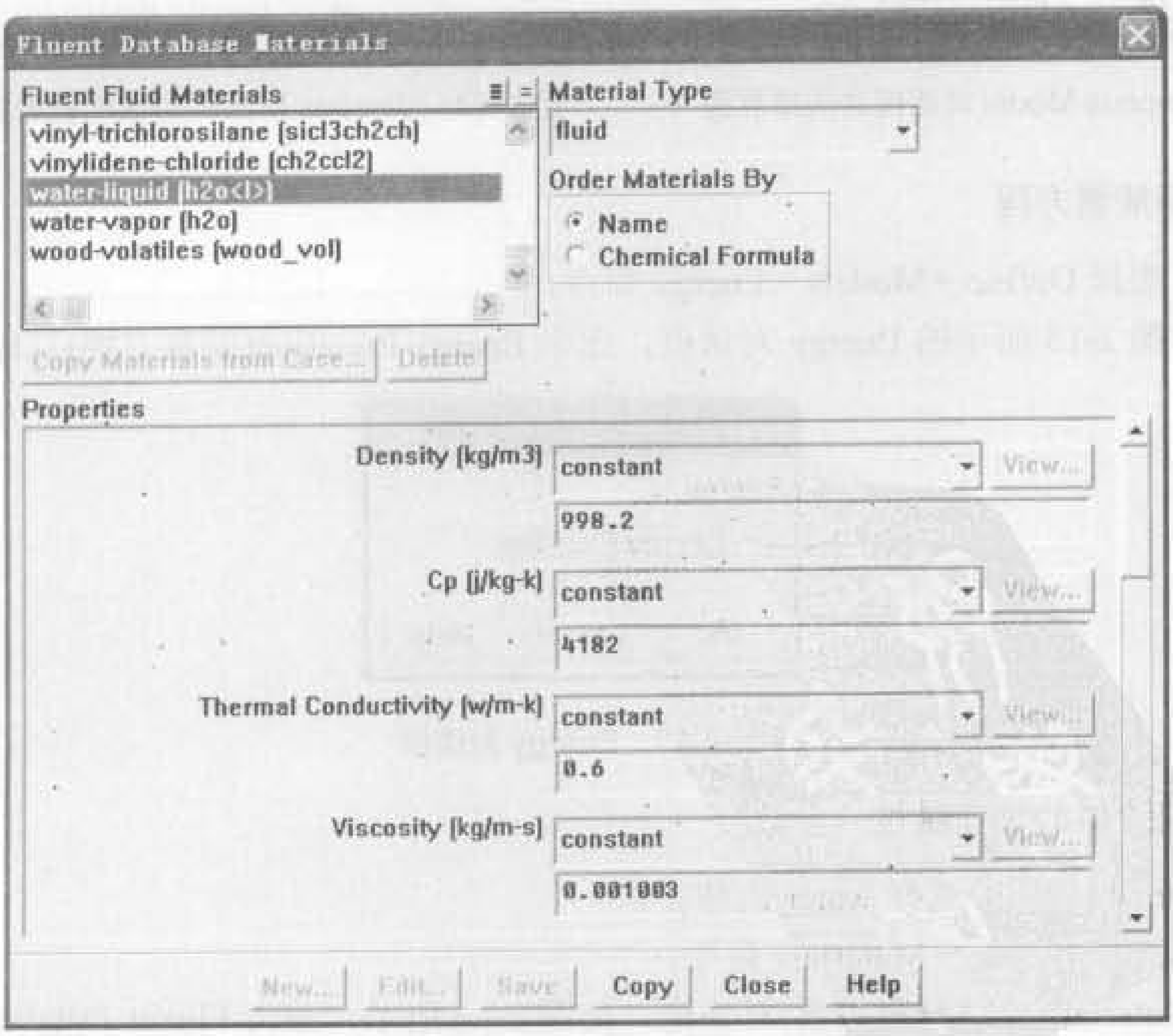


图 2-17　Fluent Database Materials(材料库选择)对话框

10. 设置边界条件

操作：选择 Define→Boundary Conditions 命令。

打开 Boundary Conditions(边界条件选择)对话框，如图 2-18 所示，图中，Zone 列表框内为区域标识，Type 列表框内为相应的属性。

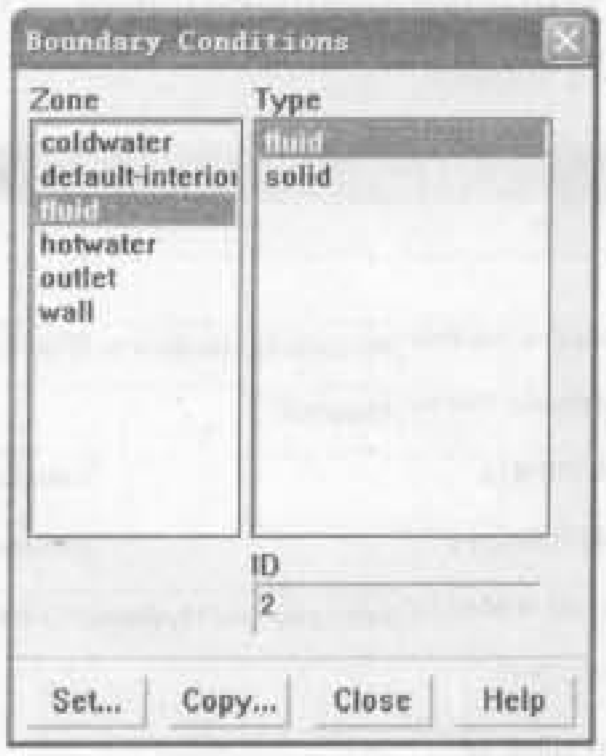


图 2-18　Boundary Conditions(边界条件选择)对话框

1)　设置流体

在 Zone(区域)列表框中选择 fluid，在 Type 列表框内显示其类型为 fluid；单击 Set 按钮，打开 Fluid(流体设置)对话框，如图 2-19 所示；在 Material Name(材料名称)下拉列表框中选择 water-liquid，单击 OK 按钮关闭 Fluid 对话框。

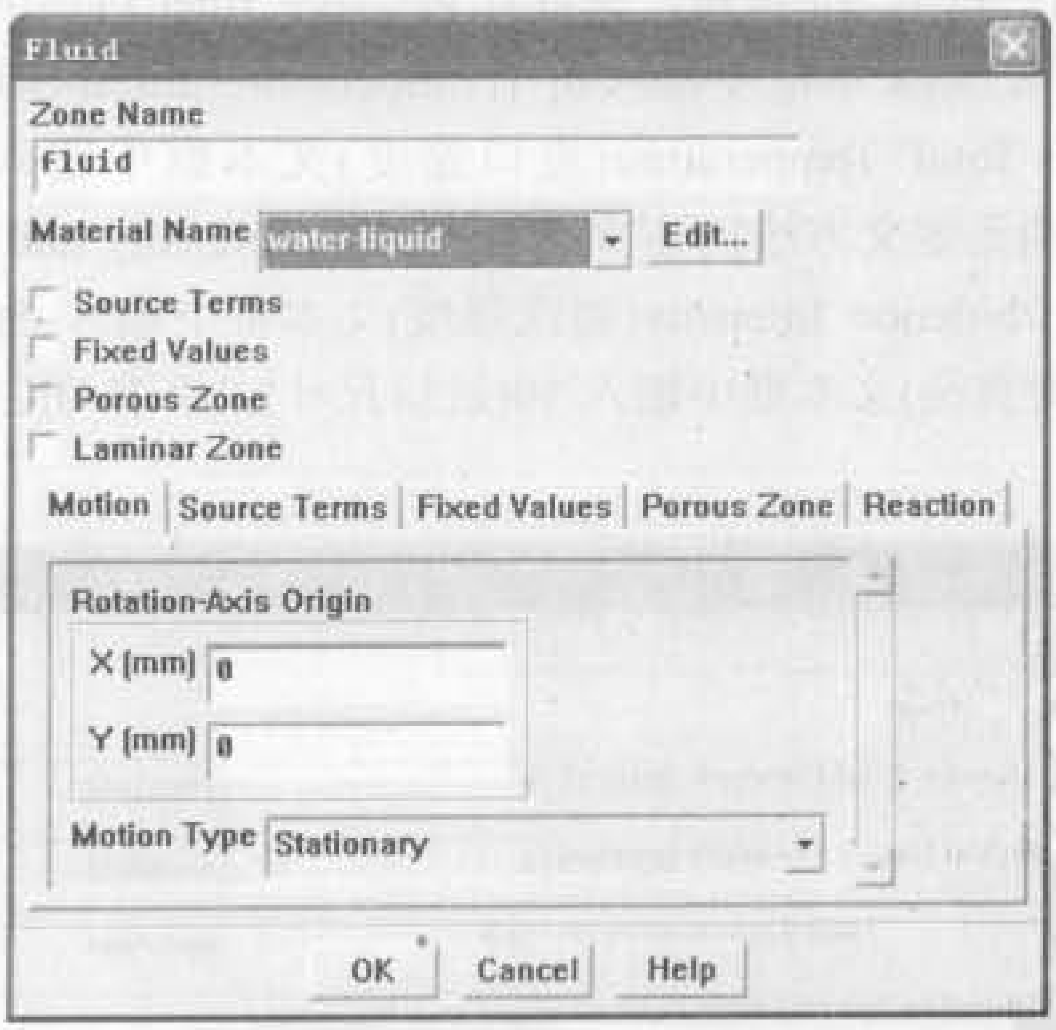


图 2-19　Fluid(流体设置)对话框

2)　设置冷水进口速度边界条件

在如图 2-18 所示的对话框中的 Zone 列表框中选择 coldwater，则在 Type 列表框内显示其类型为 velocity_inlet；单击 Set 按钮，则打开 Velocity Inlet(速度边界设置)对话框，如图 2-20 所示；在 Velocity Specification Method(速度给定方式)下拉列表框中选择 Magnitude，

Normal to Boundary(给定速度大小，速度方向垂直于边界)；在 Velocity Magnitude(进口速度)文本框中输入 6，在其右侧的下拉列表框中选择 constant(常值)；在 Temperature(进口温度)文本框中输入 298；在 Turbulence Specification Method (湍流定义方法)下拉列表框中选择 Intensity and Hydraulic Diameter(强度与水力直径)；在 Turbulence Intensity(湍流强度)文本框中输入 5(来流的湍流强度)；在 Hydraulic Diameter(水力直径)文本框中输入 30(进口尺寸)；单击 OK 按钮关闭 Velocity Inlet 对话框。

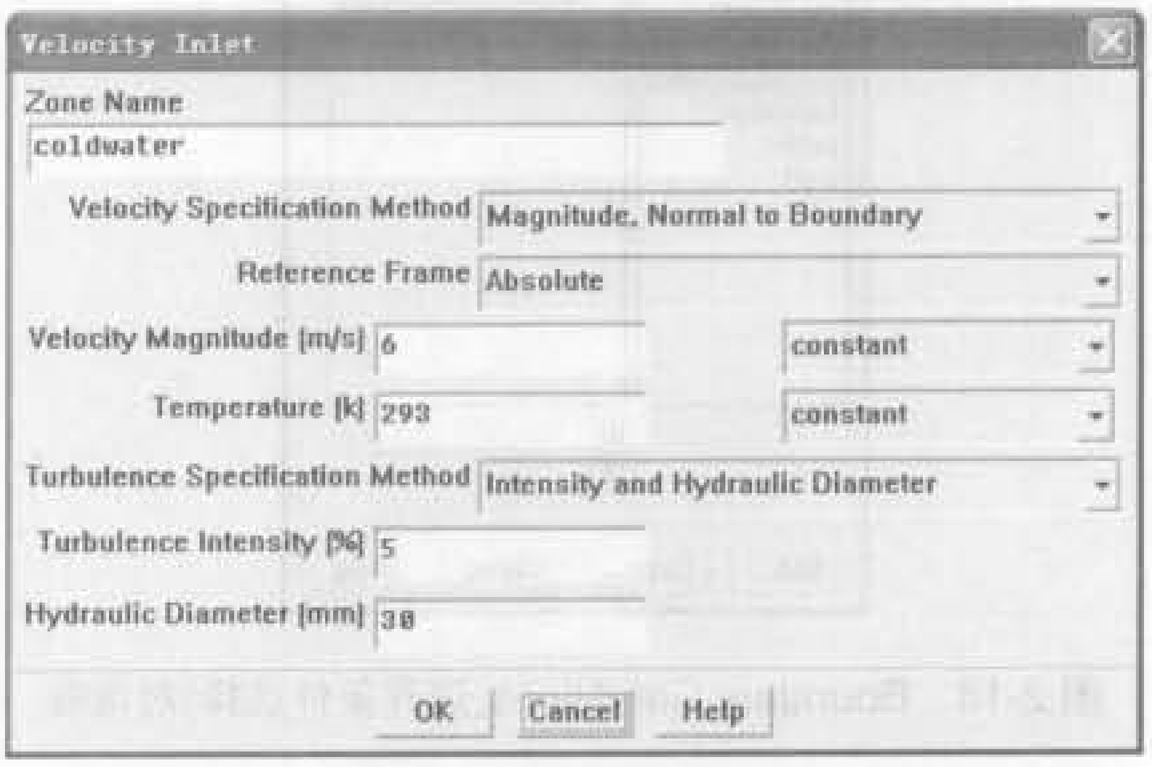


图 2-20 Velocity Inlet(速度边界设置)对话框

3) 设置热水进口压力边界条件

在如图 2-18 所示的对话框中的 Zone 列表框中选择 hotwater，则在 Type 列表框内显示其类型为 pressure_inlet；单击 Set 按钮，则打开 Pressure Inlet 对话框，如图 2-21 所示；在 Gauge Total Pressure(总表压)文本框中输入 0；在 Supersonic/Initial Gauge Pressure(初始表压)文本框中输入 0；在 Total Temperature(进口温度)文本框中输入 360；在 Turbulence Specification Method (湍流定义方法)下拉列表框中选择 Intensity and Hydraulic Diameter(强度与水力直径)；在 Turbulence Intensity(湍流强度)文本框中输入 5(来流的湍流强度)；在 Hydraulic Diameter(水力直径)文本框中输入 10(进口尺寸)；单击 OK 按钮关闭 Pressure Inlet 对话框。

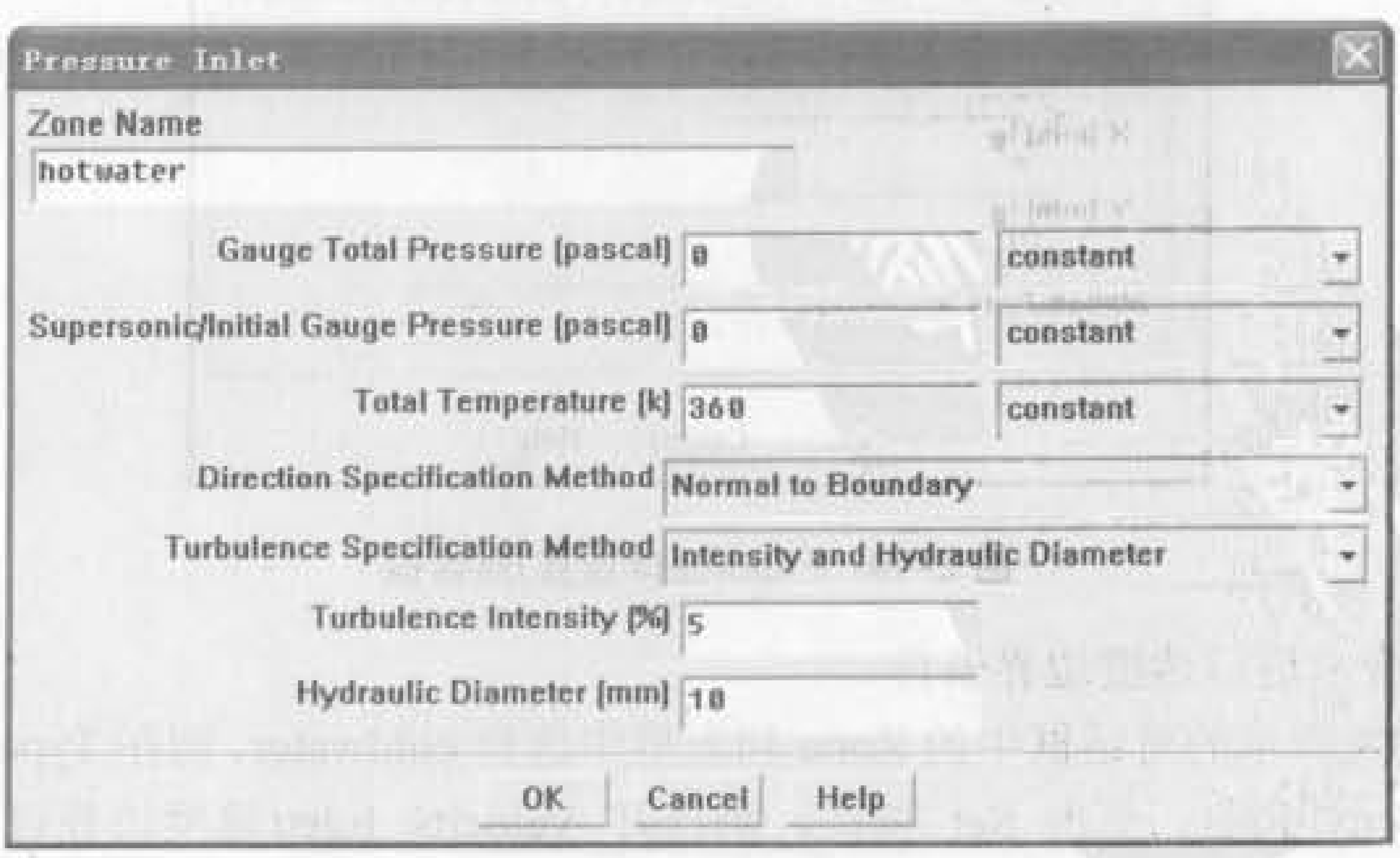


图 2-21 Pressure Inlet(压力进口边界设置)对话框

4)　设置出口压力边界条件

在如图 2-18 所示的对话框中的 Zone 列表框中选择 outlet，则在 Type 列表框内显示其类型为 pressure-outlet；单击 Set 按钮，则打开 Pressure Outlet(压力出口边界设置)对话框，如图 2-22 所示；在 Gauge Total Pressure(总表压)文本框中输入 0；在 Supersonic/Initial Gauge Pressure(初始表压)文本框中输入 0；Temperature(进口温度)文本框保持默认值；在 Turbulence Specification Method (湍流定义方法)下拉列表框中选择 Intensity and Hydraulic Diameter(强度与水力直径)；在 Turbulence Intensity(湍流强度)文本框中输入 5(来流的湍流强度)；在 Hydraulic Diameter(水力直径)文本框中输入 30(出口尺寸)；单击 OK 按钮关闭 Pressure Outlet 对话框。

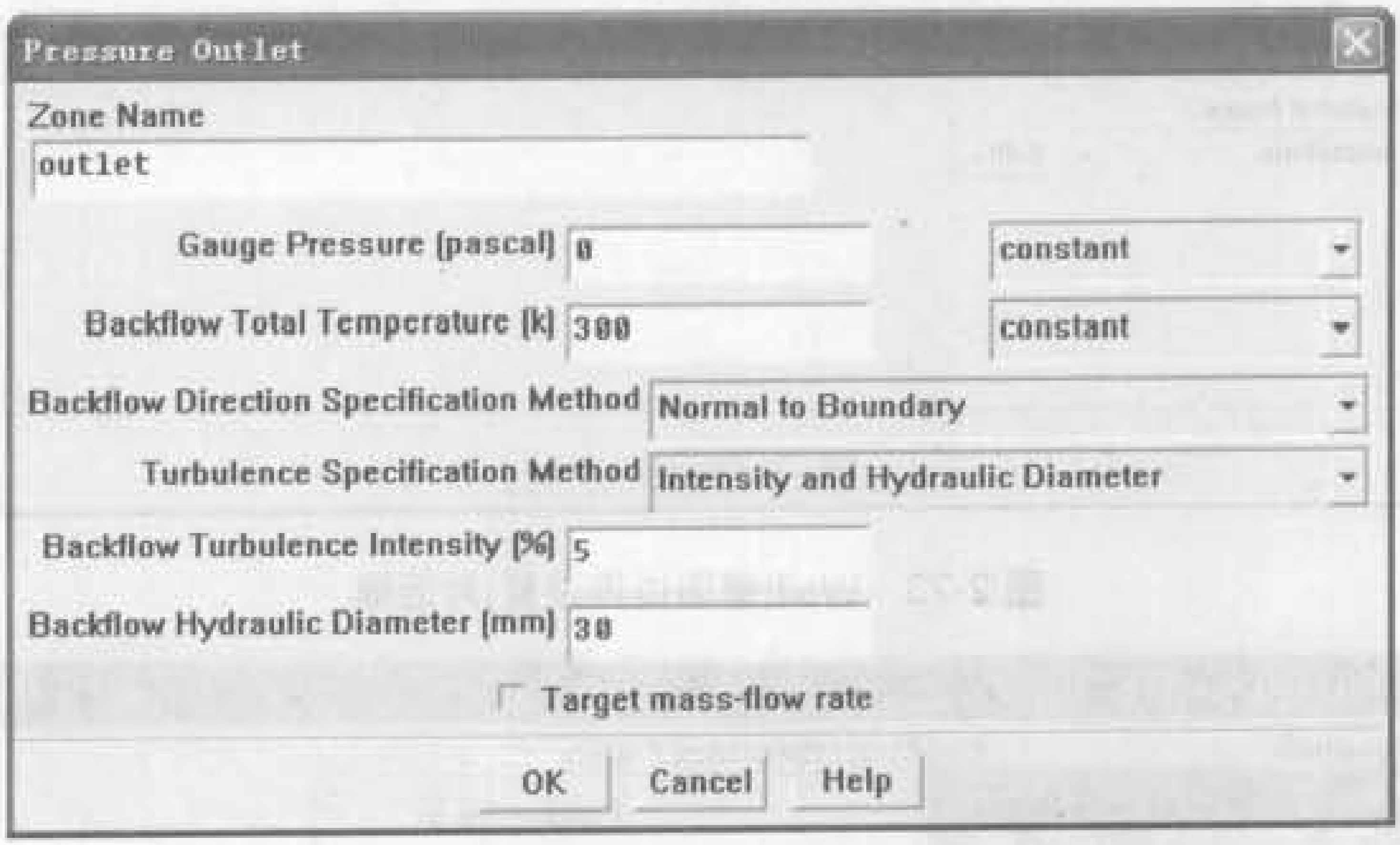


图 2-22　Pressure Outlet(压力出口边界设置)对话框

注意： 对于出口未知的情况，常设为 Outflow 边界类型，因本例中有压力进口，不能采用 Outflow 边界，所以只能采用 Pressure-outlet 边界。

5)　壁面设置

对于壁面边界条件，可以设置热力学、运动、组分、自定义源项等，本例保持默认设置，如图 2-23 所示。

11. 求解设置

操作：选择 Solve→Controls→Solution 命令。

打开 Solution Controls(求解控制设置)对话框，如图 2-24 所示，Equations 选项组中显示需要求解的方程；Under-Relaxation Factors 选项组中为各项的松弛因子的设定；Pressure-Velocity Coupling 选项组中为压力速度耦合算法的设定，有 SIMPLE、SIMPLEC、PISO 三种算法供选择；Discretization 选项组中为各项的离散格式选项。保持默认设置，单击 OK 按钮关闭对话框。

12. 求解监视设置

操作：选择 Solve→Monitors→Residual 命令。

打开 Residual Monitors(残差监视器设置)对话框，如图 2-25 所示，监视器输出方式在

Options 选项组中选择，Print 表示在 Fluent 控制台窗口中打印输出，Plot 表示在图形窗口以残差曲线的形式输出；修改收敛判据，保持默认设置，单击 OK 按钮关闭对话框。

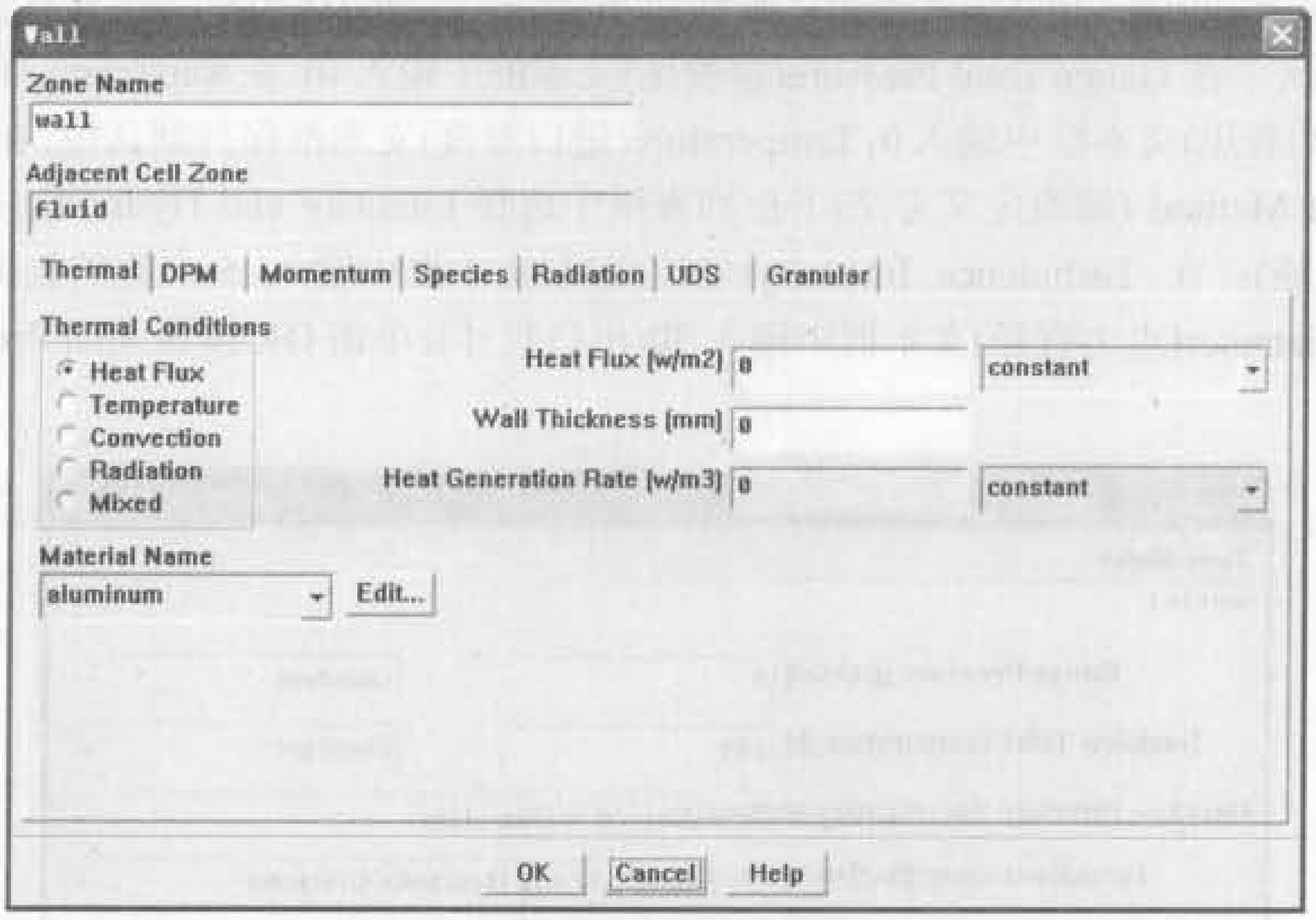


图 2-23 Wall(壁面边界设置)对话框

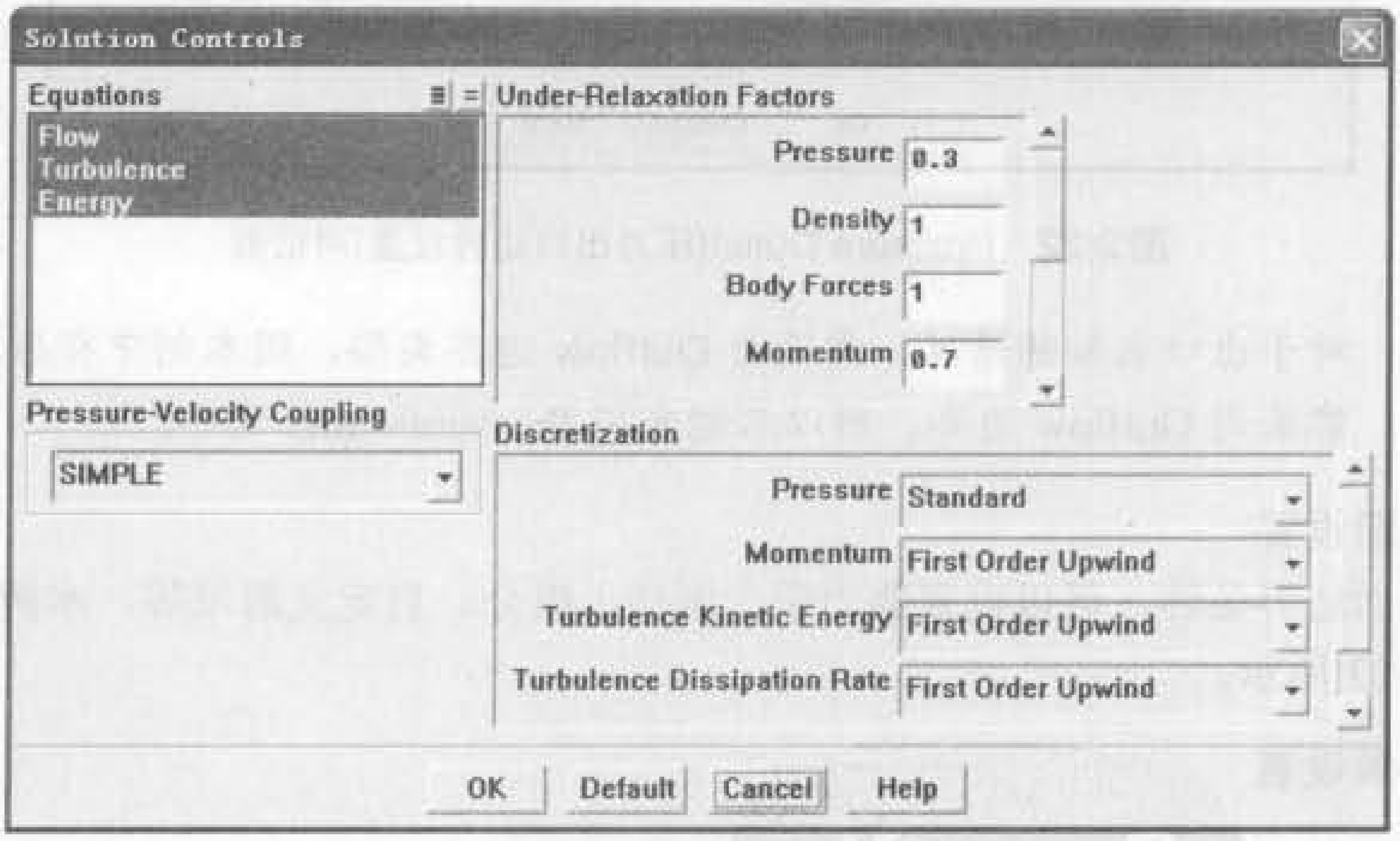


图 2-24 Solution Controls(求解控制设置)对话框

为了更好地判断计算收敛，需对所关心的截面上的物理量进行监测。本例对出口的面积平均温度进行监测，图 2-26 是 Surface Monitors(表面监视器)对话框，在该对话框中，将 Surface Monitors 微调框中的数目增加到 1；选中 Plot(若同时选中 Write，还可将结果写入文件)；单击 monitor-1 最右边的 Define 按钮，则出现 Define Surface Monitor(表面监视器定义)对话框，如图 2-27 所示；在 Report of 选项组中选择 Temperature 和 Static Temperature；在 Surface 选项组中选择监测表面为 outlet；在 Report Type 下拉列表框中选择 Area-Weighted Average(面积平均)；单击 OK 按钮；单击 Surface Monitors 对话框中的 OK 按钮。

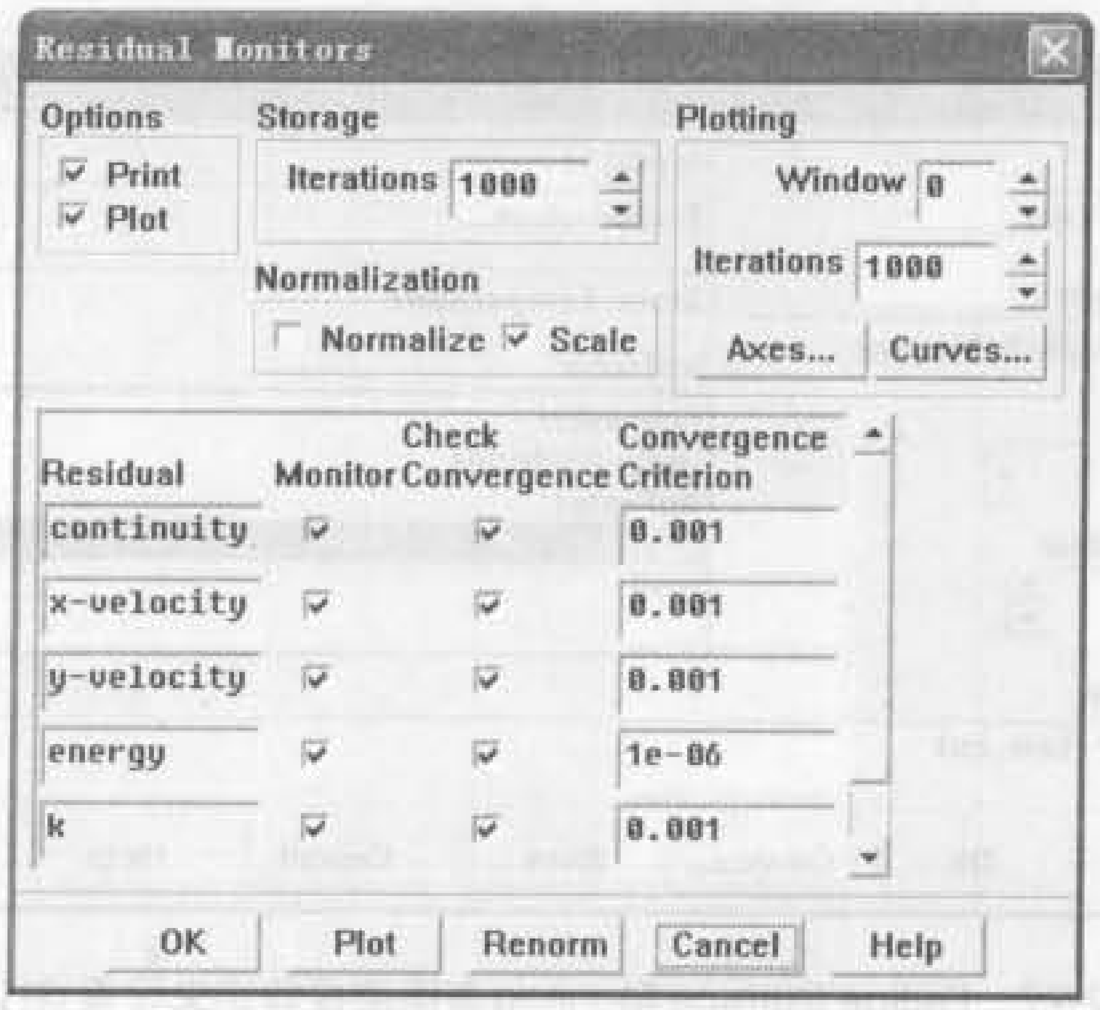


图 2-25　Residual Monitors(残差监视器设置)对话框

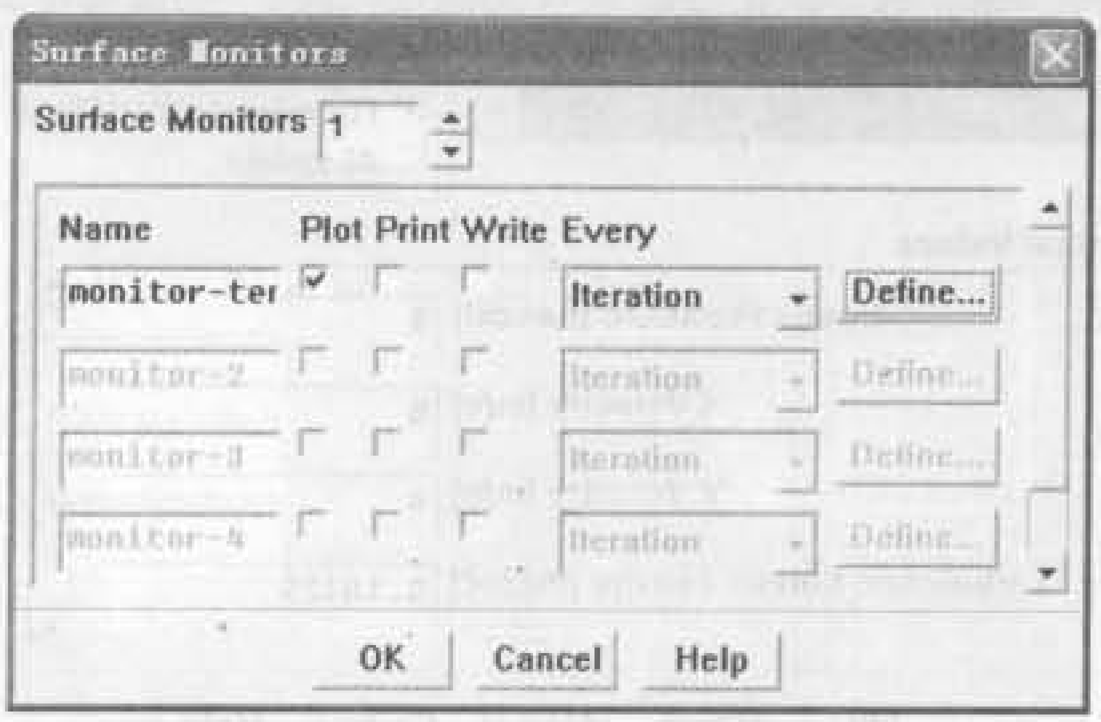


图 2-26　Surface Monitors(表面监视器设置)对话框

13. 流场初始化

操作：选择 Solver→Initialize→Initialize 命令。

打开 Solution Initialization(初始化设置)对话框，如图 2-28 所示，在 Compute From 下拉列表框中选择 all-zones，其余设置保持默认状态，对所有区域进行统一初始化。

14. 保存 case 文件(jet-reactor.cas)

操作：选择 File→Write→Case 命令。

在打开的对话框中输入文件名，单击 OK 按钮。

15. 迭代计算

操作：选择 Solver→Iterate 命令。

打开 Iterate(迭代参数设置)对话框，如图 2-29 所示，在 Number of Iterations(迭代次数)微调框中输入 500，单击 Iterate 按钮开始计算。

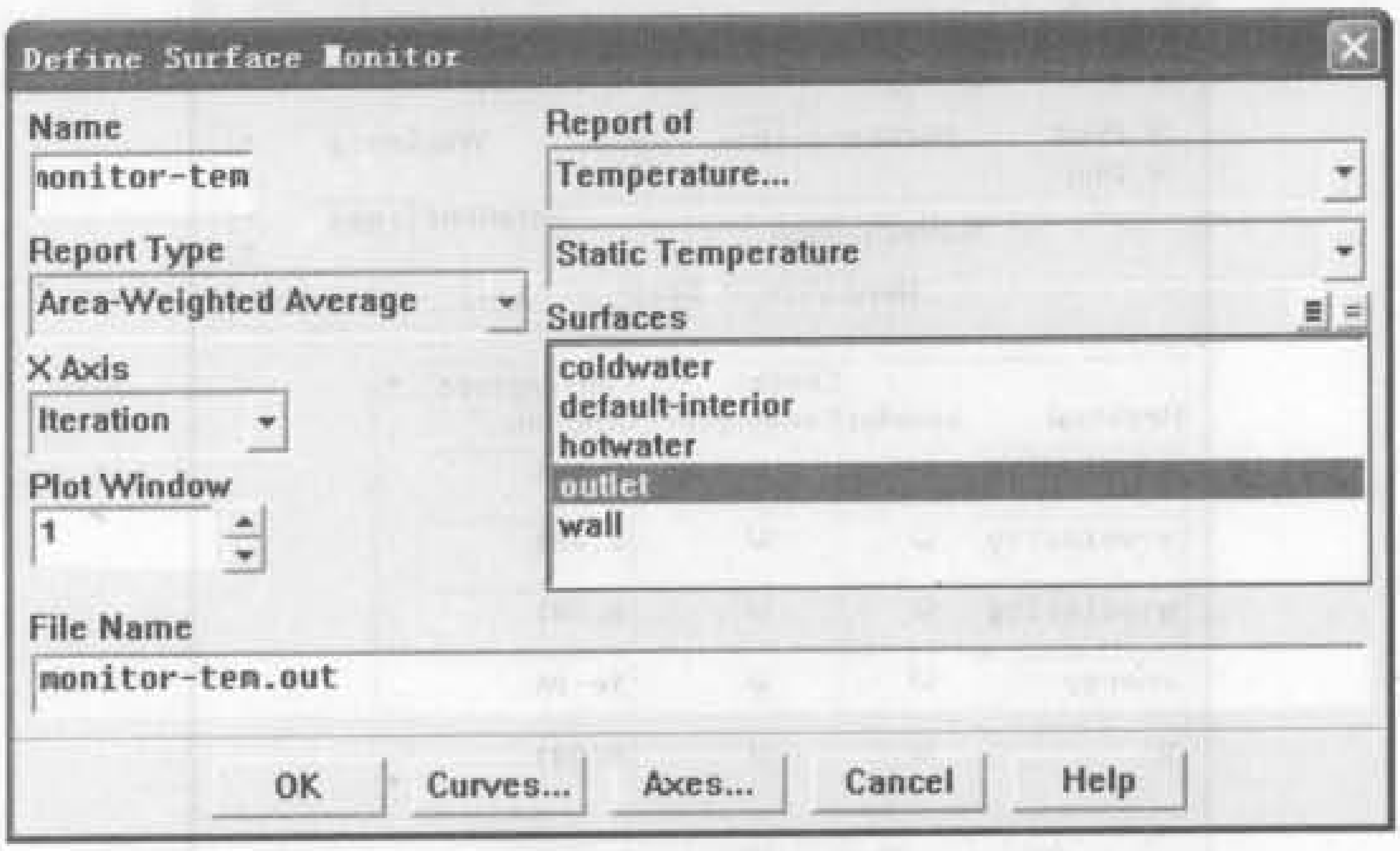


图 2-27 Define Surface Monitor(表面监视器定义设置)对话框

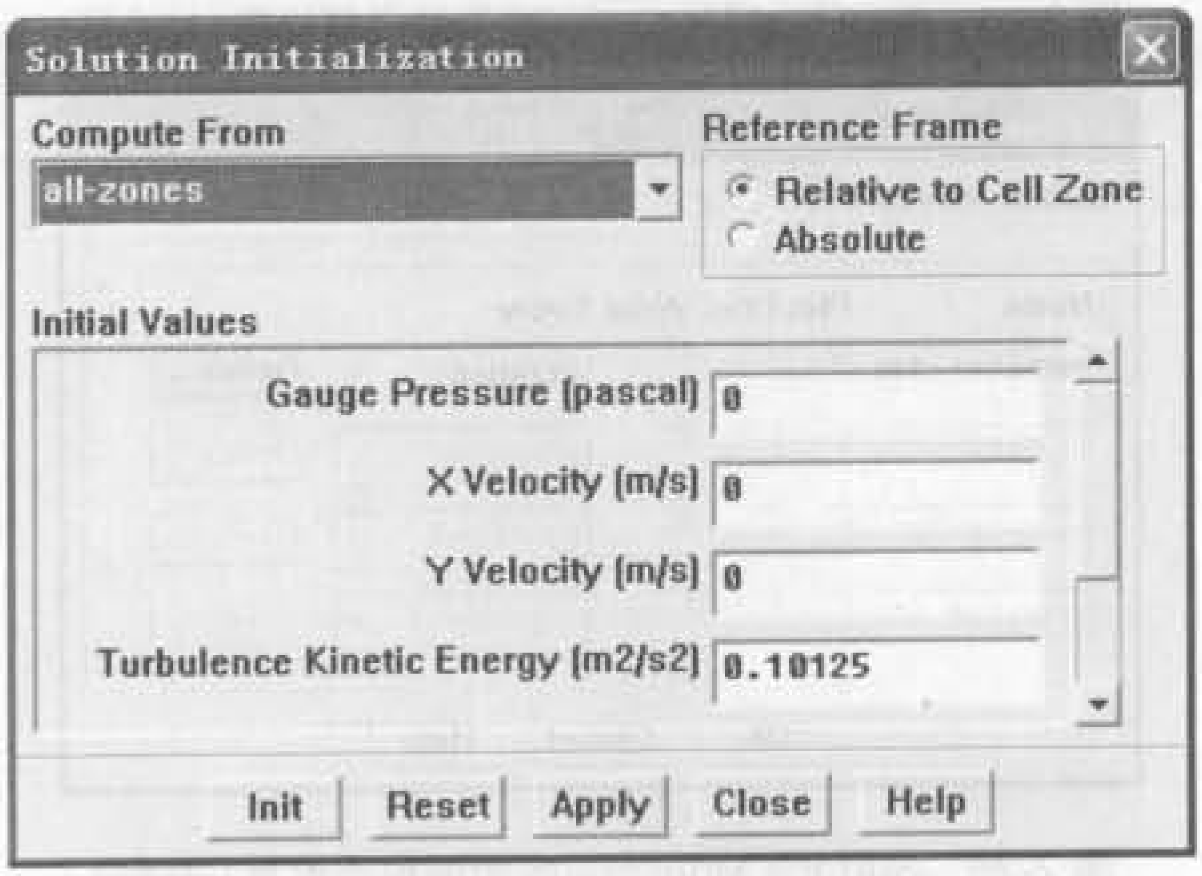


图 2-28 Solution Initialzation(初始化设置)对话框

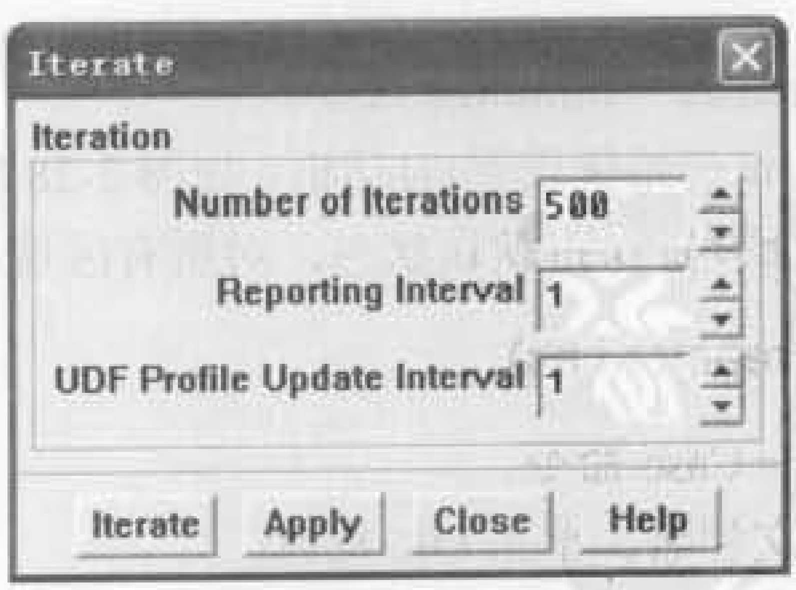


图 2-29 Iterate(迭代参数设置)对话框

进行 216 次迭代后，残差监视器显示收敛，如图 2-30 所示；出口截面上的平均温度监视器窗口显示的曲线如图 2-31 所示，形成一条水平线。

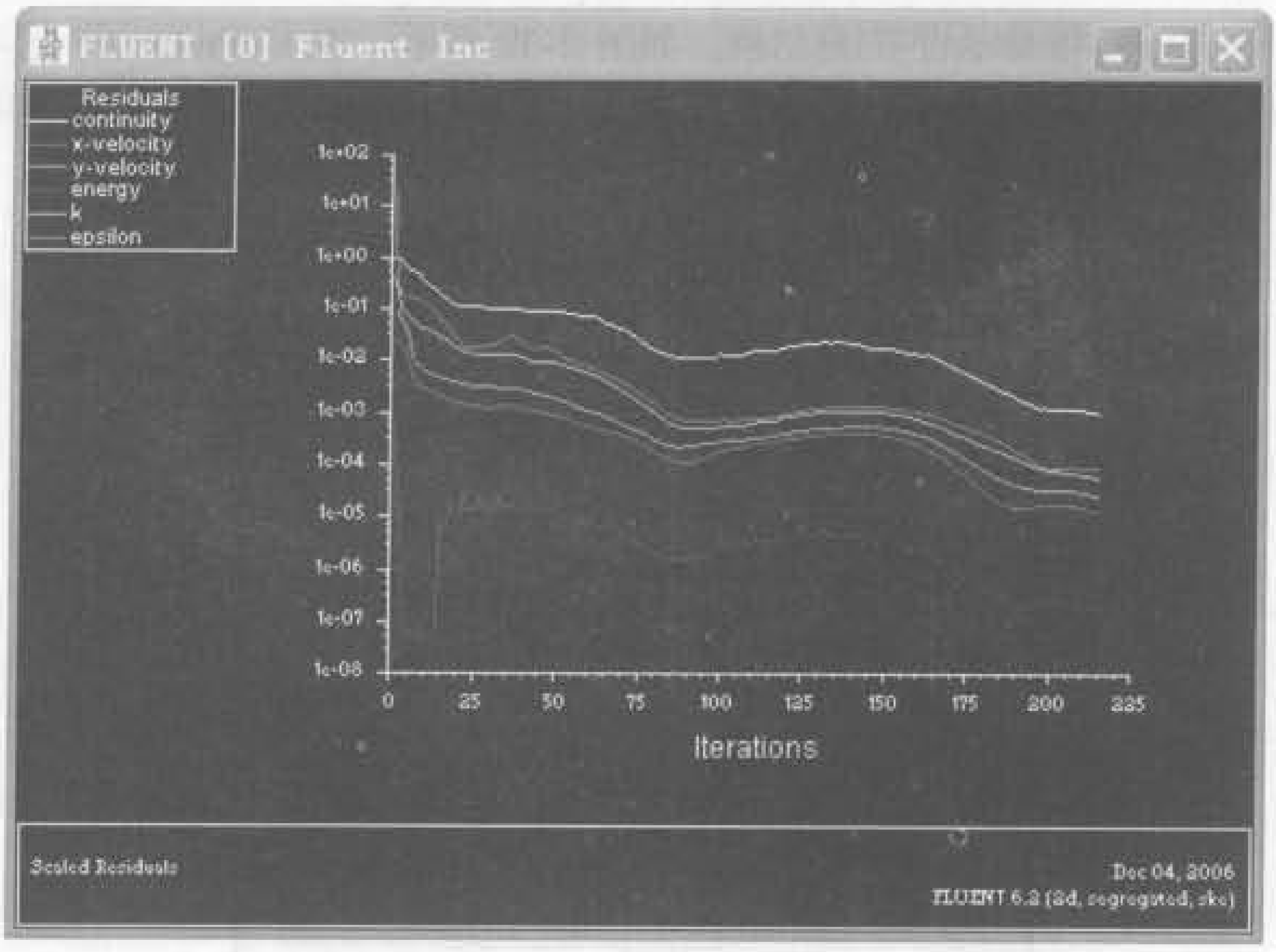


图 2-30　残差曲线图

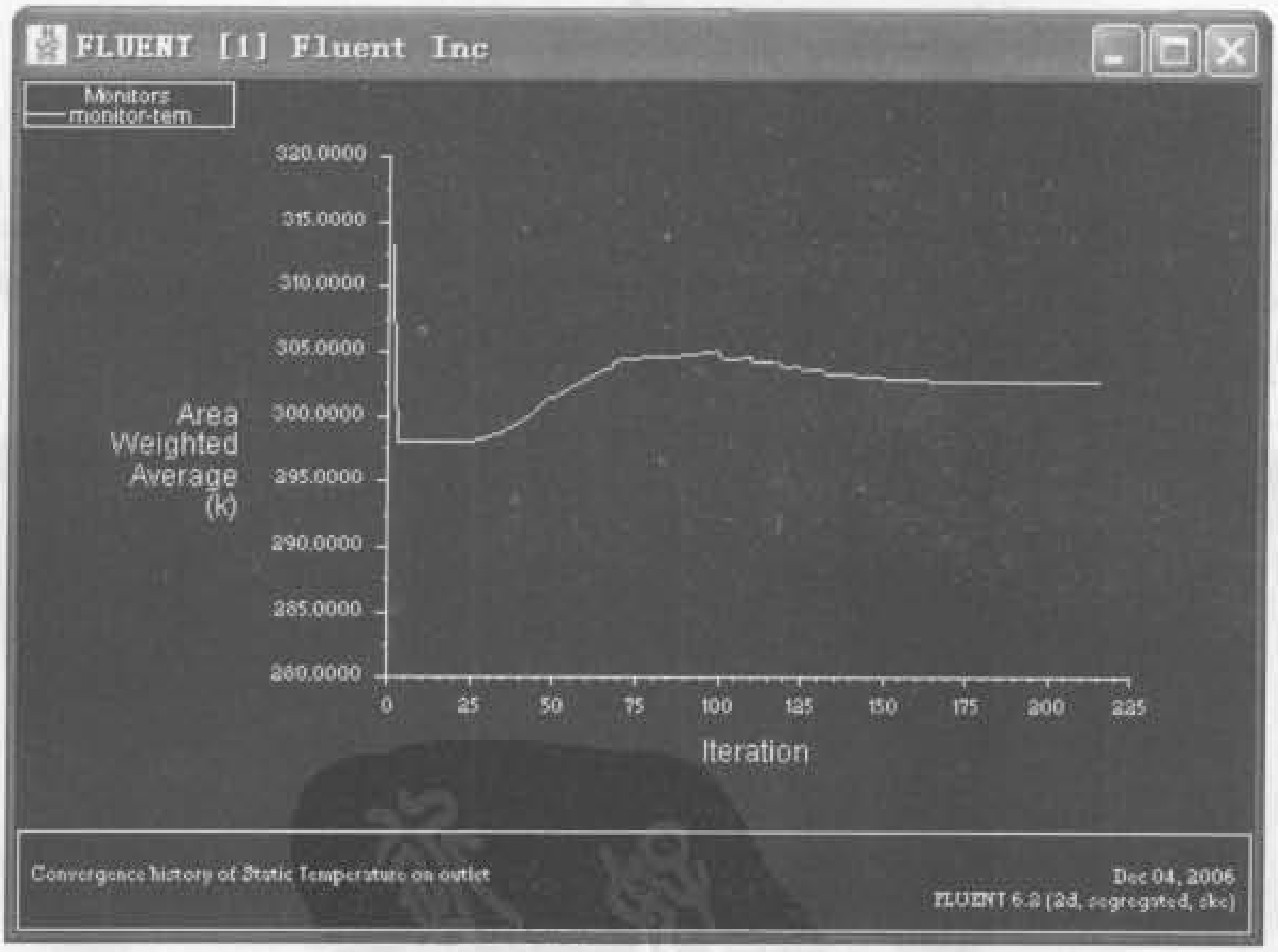


图 2-31　出口温度曲线图

16. 保存 data 文件(jet-reactor.dat)

操作：选择 File→Write→Data 命令。

在打开的对话框的 Data File 文本框中输入文件名，单击 OK 按钮。

17. 显示温度云图

操作：选择 Display→Contours 命令。

打开 Contours(等值显示设置)对话框，如图 2-32 所示，在 Contours of 选项组中选择 Temperature(温度)和 Static Temperature(静态温度)；在 Options 选项组中选中 Filled(填充方式)复选框；单击 Display 按钮，将显示如图 2-33 所示的计算域中温度场分布。



图 2-32 Contours(等值显示设置)对话框

18. 显示压力云图

在如图 2-32 所示的 Contours 对话框中的 Contours of 选项组中选择 Pressure (压力)和 Static Pressure(静态压力)；在 Options 选项组中选中 Filled(填充方式)复选框；单击 Display 按钮，将显示如图 2-34 所示的计算域中压力场分布。

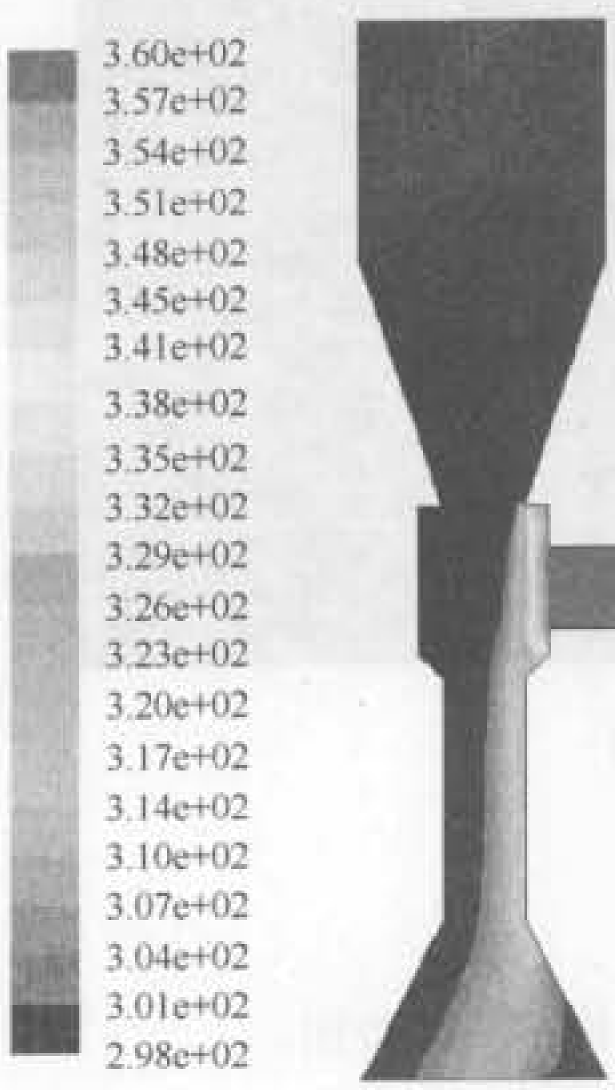


图 2-33 温度云图

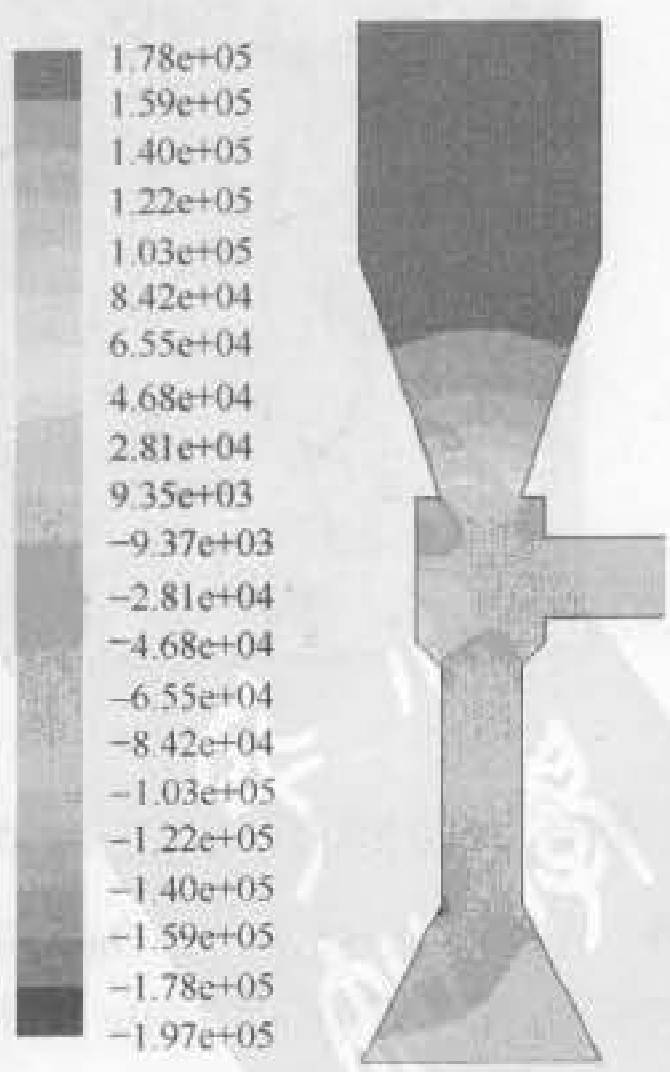


图 2-34 压力云图

19. 显示速度云图

在如图 2-32 所示的 Contours 对话框中的 Contours of 选项组中选择 Velocity(速度)和 Velocity Magnitude(速度大小)；在 Options 选项组中选中 Filled(填充方式)复选框；单击 Display 按钮，将显示如图 2-35 所示的计算域中速度场分布。

20. 显示速度矢量图

操作：选择 Display→Velocity Vectors 命令。

打开 Vectors(速度矢量设置)对话框，如图 2-36 所示，保持默认设置，单击 Display 按钮，则将显示如图 2-37 所示的计算域中速度矢量图。

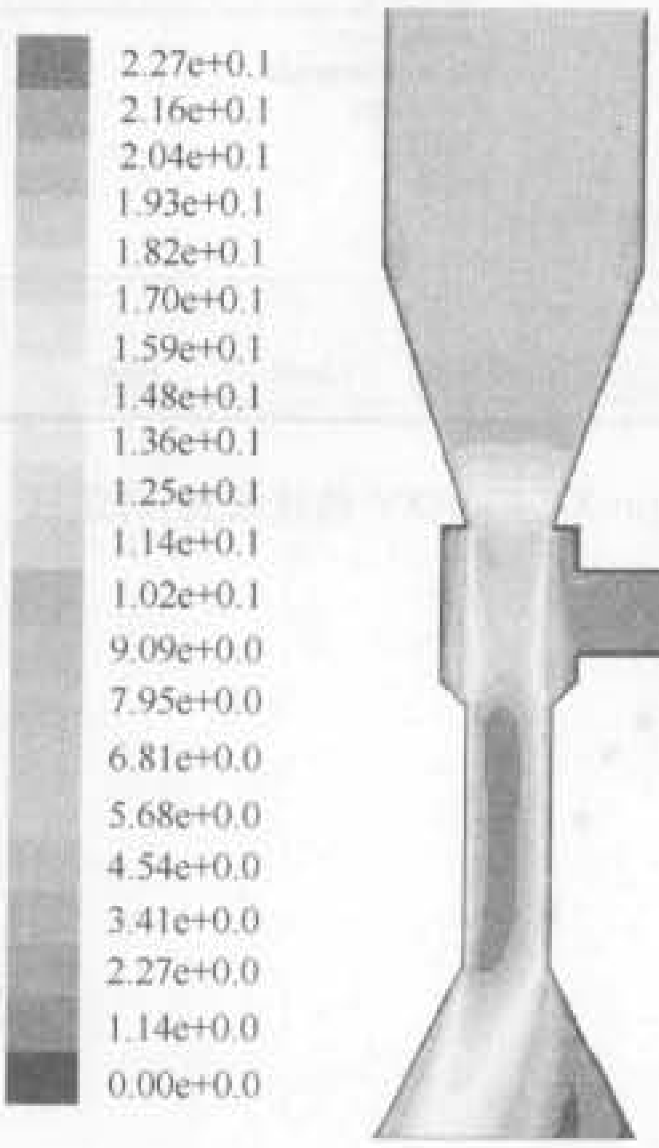


图 2-35　速度云图

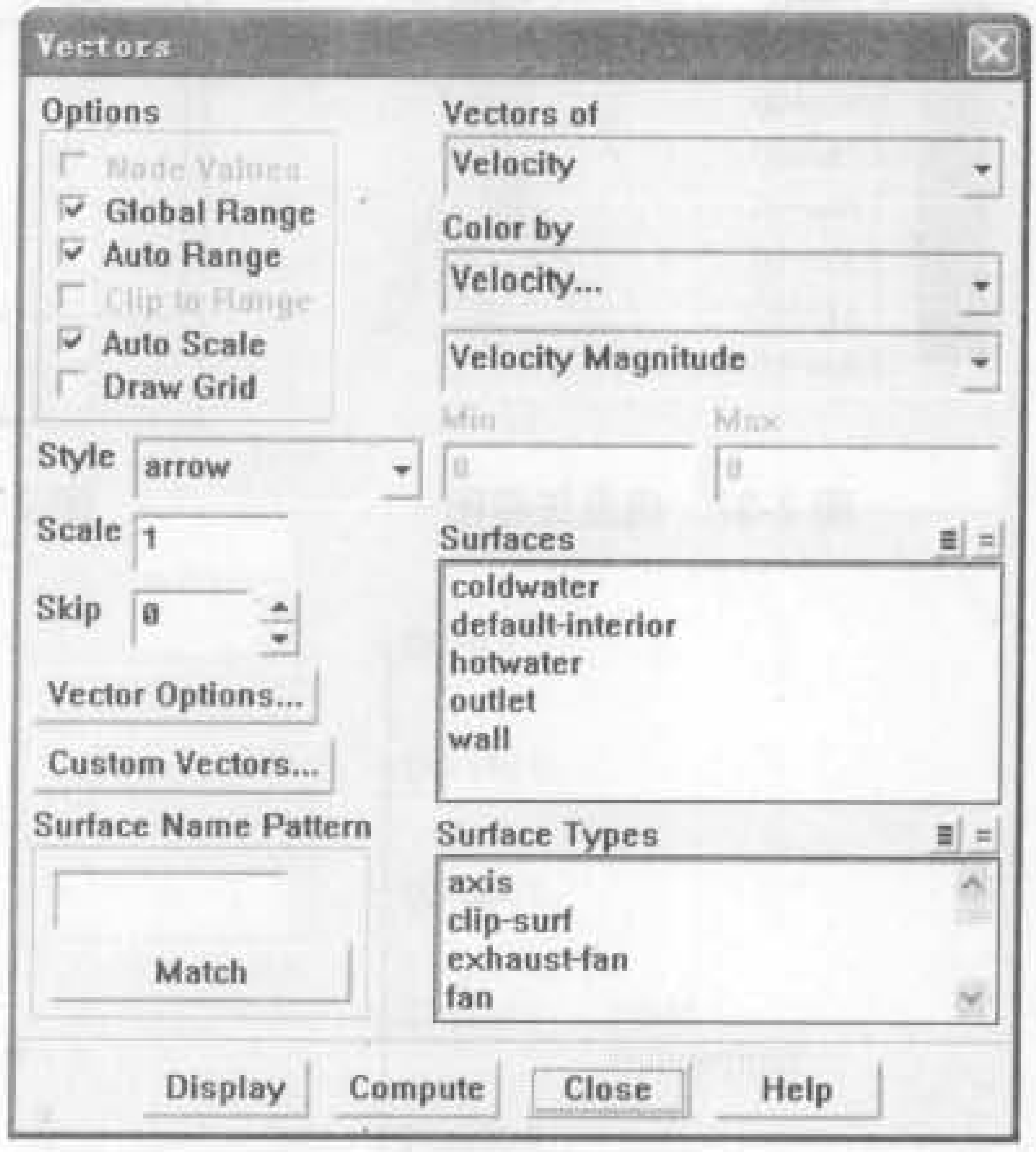


图 2-36　Vectors(速度矢量设置)对话框

21. 显示出口截面温度 XY 曲线图

操作：选择 Plot→XY Plot 命令。

打开 Solution XY Plot(XY 曲线设置)对话框，如图 2-38 所示，在 Y Axis Function(Y 轴函数)选项组中选择 Temperature 和 Static Temperature；在 Surfaces(表面)选项组中选择 outlet，其他保持默认设置，单击 Plot 按钮，则将显示如图 2-39 所示的出口截面上的温度分布图。

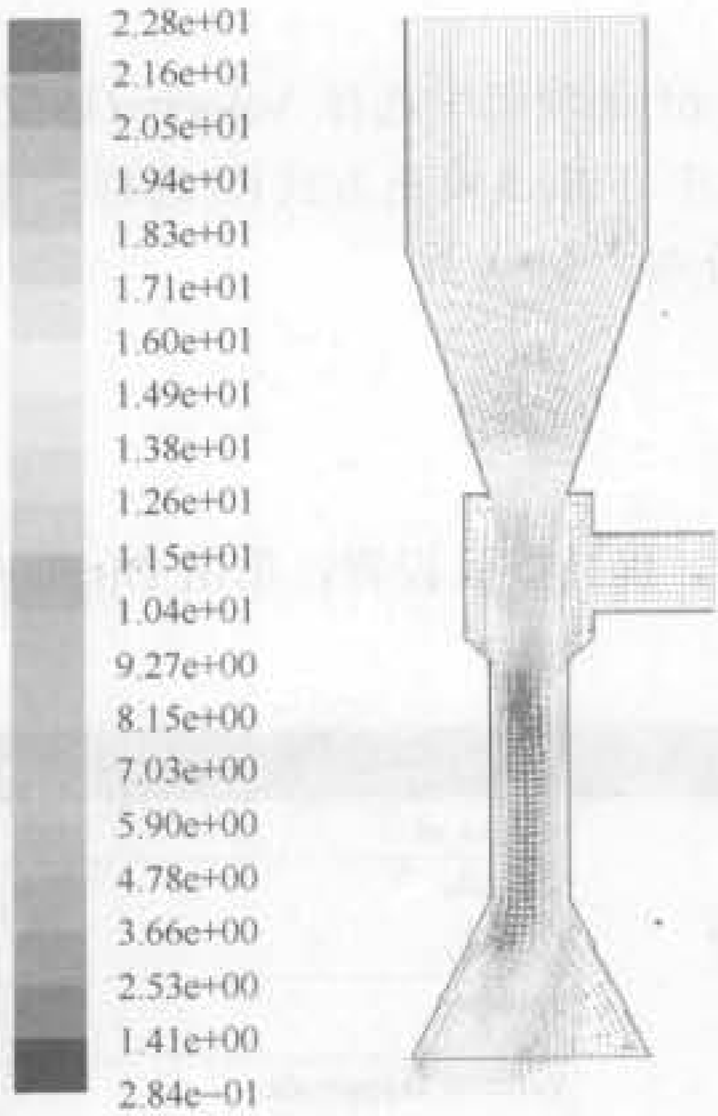


图 2-37 速度矢量图

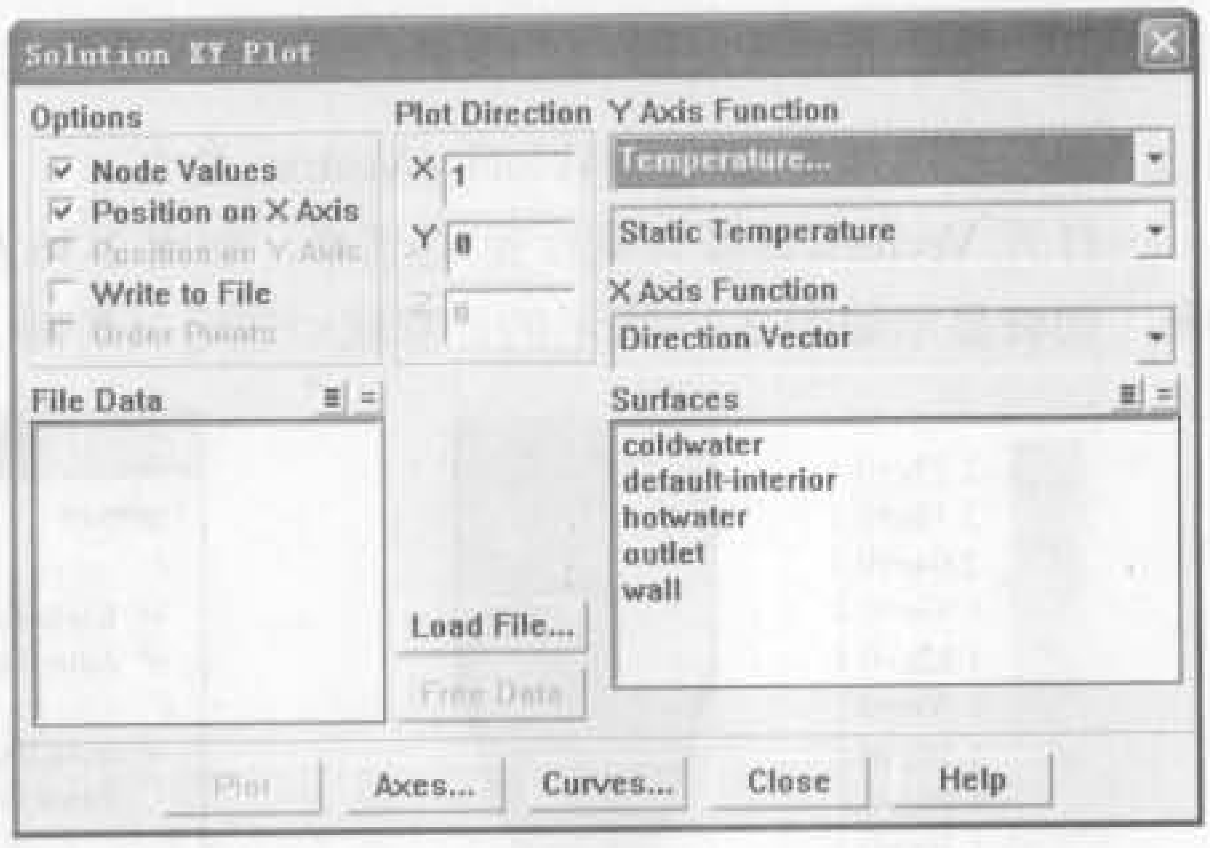


图 2-38 Solution XY Plot(XY 曲线设置)对话框

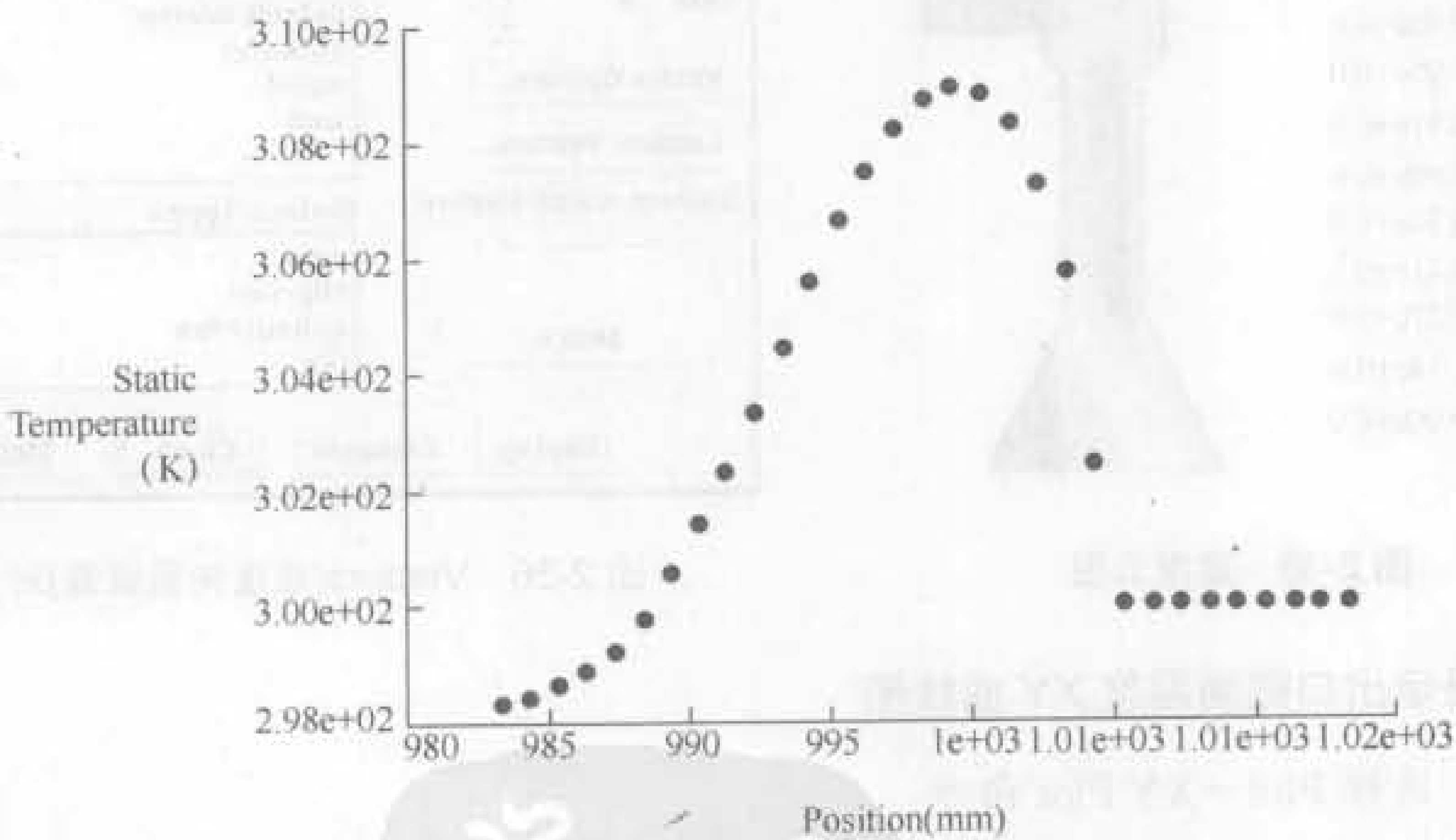


图 2-39 出口截面上温度分布图

注意： Fluent 中还有很多后处理方式，可根据自己的需要进行计算结果的处理。

2.4 Fluent 读取 Ansys 网格的操作

Fluent 除了可以读取 Gambit 的网格文件，还可以读取其他软件的网格文件，如 Ansys、Patran、TGrid、GeoMesh、preBFC、ICEMCFD、I-DEAS、Nastran 等。这里介绍读取 Ansys 网格文件的方法及实例。

2.4.1 Fluent 读取 Ansys 网格的方法

运行 Ansys，在 Ansys 中进行如下操作。

(1) 打开 Ansys 网格文件。

(2) 输入命令：ALLSEL,ALL 或选取需要的网格和节点。

(3) 输入命令：CDWRITE,DB,yourfilename.cdb,,,。

或通过如下 GUI 方式输出网格文件：

```
Menu Paths
Main Menu>Preprocessor>Archive Model>Write
```

打开 Fluent，在 Fluent 中进行如下操作。

(1) 通过如下 GUI 方式

```
File>Import>ANSYS>Input File...
```

(2) 选取 yourfilename.cdb。

(3) 单击 OK 按钮。

2.4.2 Fluent 读取 Ansys 网格的实例

(1) 在 Ansys 中建模，并划分网格，如图 2-40 所示为河道网格模型。在命令框中输入 ALLSEL,ALL，按 Enter 键，接着再输入 CDWRITE,DB,river_msh.cdb,,,，输出网格模型文件 river_msh.cdb。

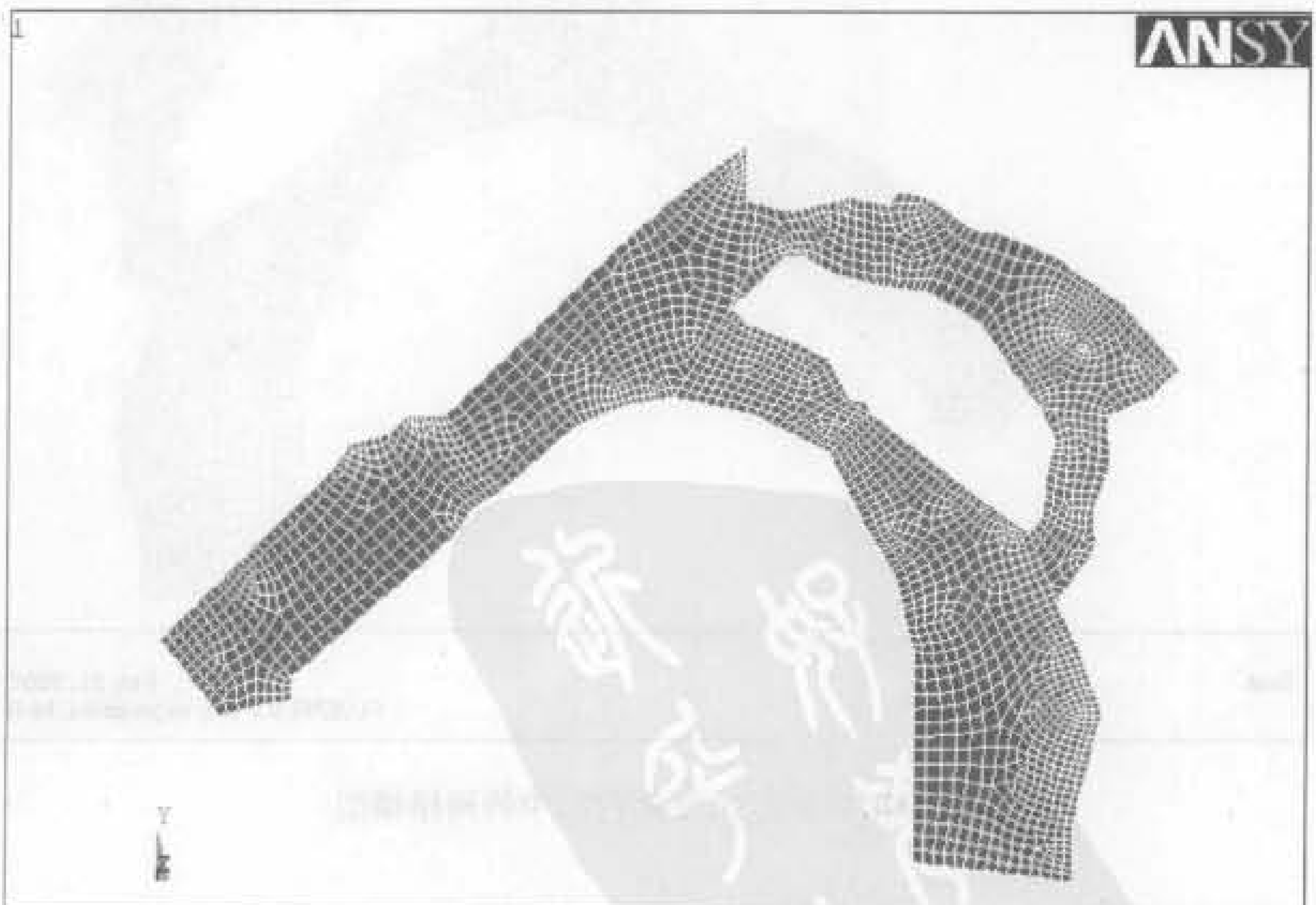


图 2-40　Ansys 中网格模型

(2) 在 Fluent 中，选择 File→Import→ANSYS→Input File 命令，打开 Select File 对话框，选择上面输出的 river_msh.cdb 文件，单击“打开”按钮，Fluent 控制台出现如

图 2-41 所示的提示，表示输入成功。

```
> Reading "| utility fe2ram -tANSYS  -d2 C:\fluentexercise2\river_mesh.cdb"...

ANSYS file to RAMPANT file

     2389 nodes.
     4072 2D interior faces, zone  3.
      474 2D wall faces, zone  4.
Read 2389 nodes, 2157 elements, 1 groups.
     2157 mixed cells, zone  1.
       10 tri elements
     2147 quad elements


Building...
      grid,
      materials,
      interface,
      domains,
      zones,
         default-exterior
         default-interior
         group-1
      shell conduction zones,
Done.
```

图 2-41　Ansys 网格模型导入后的 Fluent 控制台的提示

选择 Display→Grid 命令，打开 Grid Display 面板，单击 Display 按钮，显示与 Ansys 窗口相同的网格模型，如图 2-42 所示。

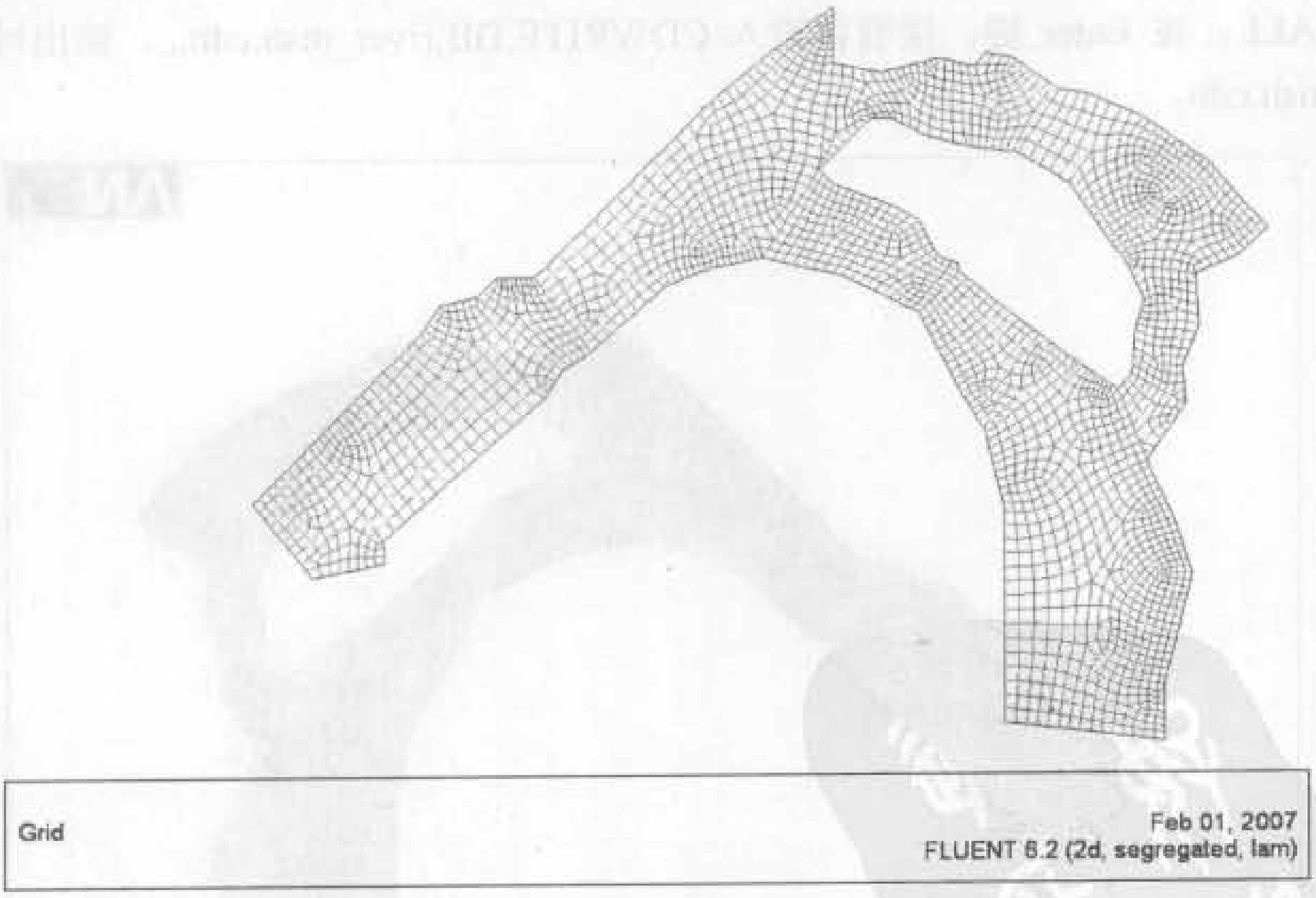


图 2-42　Fluent 图形窗口中的网格模型

第 3 章　Gambit 的使用

Gambit 是专用的 CFD 前置处理器，Fluent 系列产品皆采用 Fluent 公司自行研发的 Gambit 前处理软件来建立几何形状及生成网格，是具有超强组合建构模型能力的前处理器，由 Fluent 进行求解。也可以用 ICEM CFD 进行前处理，由 Tecplot 进行后处理。

3.1　Gambit 功能及界面

3.1.1　特点及功能

Gambit 是面向 CFD 的专业前处理器软件，它包含全面的几何建模能力，既可以在 Gambit 内直接建立点、线、面、体几何，也可以从主流的 CAD/CAE 系统如 Pro/E、UGII、I-DEAS、CATIA、SolidWorks、Ansys、Patran 导入几何体和网格，Gambit 强大的布尔运算能力为建立复杂的几何模型提供了极大的方便。

Gambit 具有灵活方便的几何修正功能，当从接口中导入几何时会自动合并重合的点、线、面；Gambit 能在保证原始几何精度的基础上通过虚拟几何自动缝合小缝隙，这样既可以保证几何精度，又可以满足网格划分的需要。

Gambit 功能强大的网格划分工具，可以划分出包含边界层等 CFD 特殊要求的高质量的网格。Gambit 中专有的网格划分算法可以保证在较为复杂的几何区域中直接划分出高质量的六面体网格。

Gambit 中的 TGrid 方法可以在极其复杂的几何区域中划分出与相邻区域网格连续的完全非结构化的网格，Gambit 网格划分方法的选择完全是智能化的，当选择一个几何区域后，Gambit 会自动选择最合适的网格划分算法，使网格划分过程变得极为容易。

Gambit 可以生成 Fluent 5/6、Fluent 4.5、FIDAP、Polyflow、Nekton、Ansys 等求解器所需要的网格。

3.1.2　基本界面

如图 3-1 所示，Gambit 用户界面可分为 7 个部分，分别为菜单栏、视图、命令面板、命令显示窗、命令解释窗、命令输入窗和视图控制面板。

1. 菜单栏

菜单栏位于操作界面的上方，有 File、Edit、Solver、Help 4 个菜单，最常用的是 File 菜单中的 New、Open、Save、Save as 和 Export 等命令。这些命令的使用和一般的软件一样。Gambit 可识别的文件后缀为.dbs，而要将 Gambit 中建立的网格模型调入 Fluent 使用，则需要将其输出为.msh 文件(选择 File→Export 命令)。

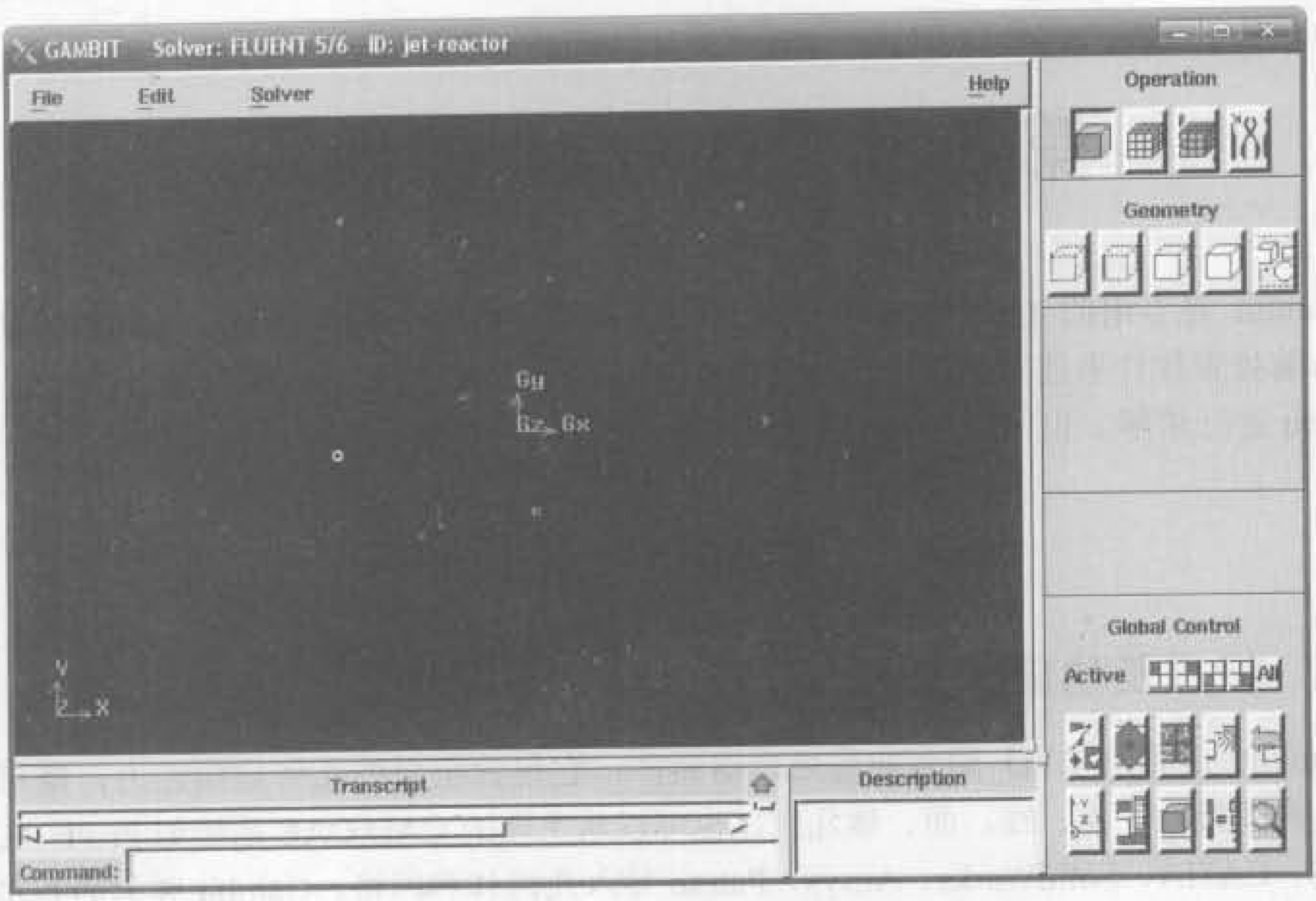


图 3-1 Gambit 的操作界面

2. 视图和视图控制面板

Gambit 中可显示 4 个视图，以便于建立三维模型；同时也可以只显示一个视图。视图的坐标轴由视图控制面板来决定。图 3-2 显示的是视图控制面板。

视图控制面板中的命令可分为两个部分，上面的一排 4 个图标表示的是 4 个视图，当激活视图图标时，视图控制面板中下方 10 个命令才会作用于该视图。

视图控制面板中常用的按钮有：全图显示、选择显示视图、选择视图坐标、选择显示项目、渲染方式。

同时，还可以使用鼠标来控制视图中的模型显示：按住左键拖动鼠标可以旋转视图，按住中键拖动鼠标则可以在视图中移动物体，按住右键上下拖动鼠标可以缩放视图中的物体。

3. 命令面板

命令面板是 Gambit 的核心部分，通过命令面板上的命令图标，可以完成绝大部分建模和网格划分的工作。图 3-3 显示的就是 Gambit 的命令面板。

从命令面板中可以看出，网格划分的工作可分为三个步骤：一是建立模型，二是划分网格，三是定义边界。这三个部分分别对应着 Operation 选项组中的前三个命令按钮 Geometry(几何体)、Mesh(网格)和 Zones(区域)。Operation 选项组中的第四个命令按钮 Tools 则是用来定义视图中的坐标系统，一般取默认值。命令面板中的各个按钮的含义和使用方法将在以后的具体例子中介绍。

4. 命令显示窗和命令输入窗

命令显示窗和命令输入窗位于 Gambit 的左下方，如图 3-4 所示。

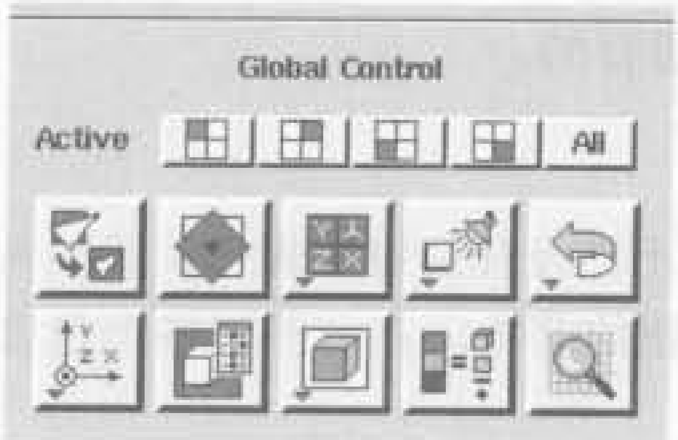


图 3-2　视图控制面板

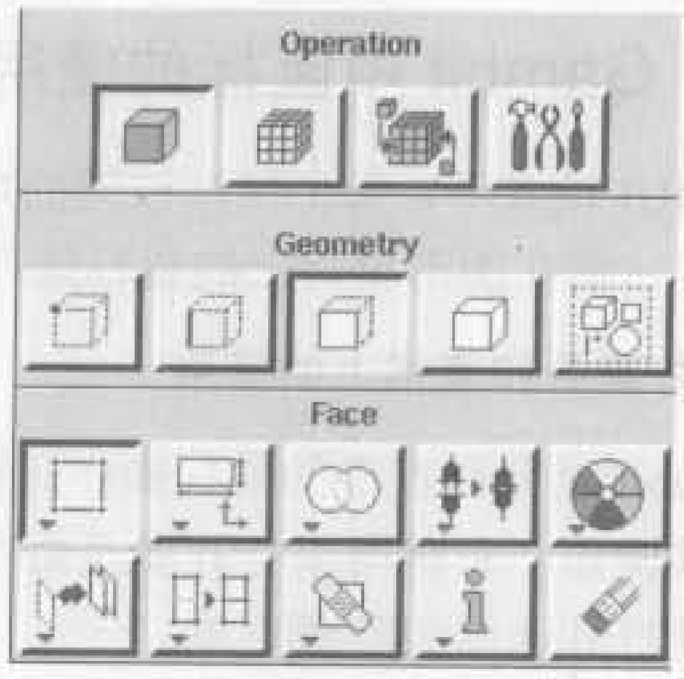


图 3-3　命令面板

命令显示窗中记录了每一步操作的命令和结果，而命令输入窗则可以直接输入命令，其效果和单击命令按钮一样。

5. 命令解释窗

图 3-5 显示的是位于命令显示窗右方的命令解释窗，当将鼠标指针放在命令面板中任意一个按钮的上面，Description 窗口中将出现对该命令的解释。

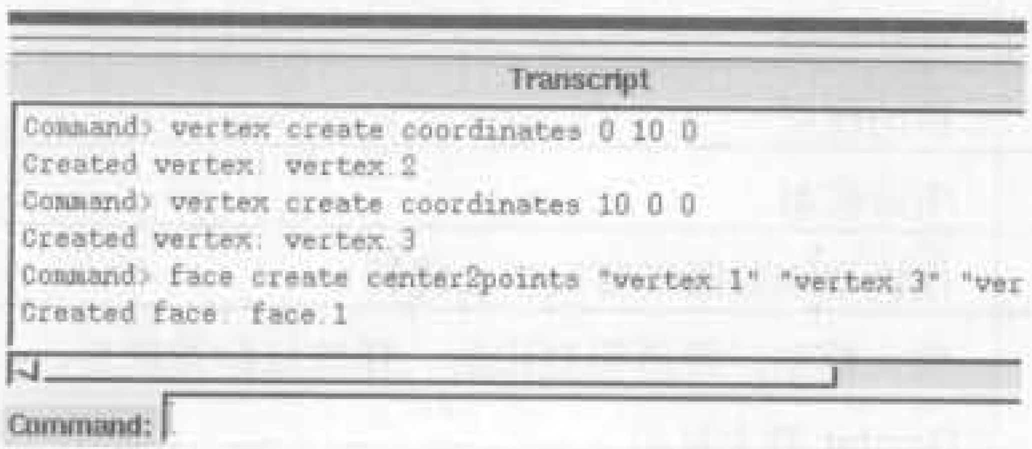


图 3-4　命令显示窗和命令输入窗

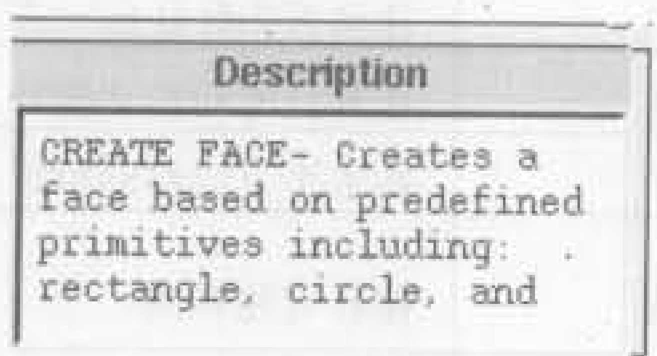


图 3-5　命令解释窗

3.1.3 Gambit 的文件组成

Gambit 通过 3 个数据文件，沿片段操作的步骤以及模型现在的状态进行。这些片段数据文件的名字、标题、格式和内容如表 3-1 所示。

表 3-1　Gambit 的文件格式

名　称	标　题	格　式	内　容
jou	日志	文本	片段中的几何体、网格、区域和执行的工具命令的连续列表
trn	文本	文本	片段中 Gambit transcript 窗口中的一系列信息
dbs	数据库	二进制	包括几何体、网格、展示、默认以及模型有关的日志消息的二进制数据库

除上面列的 3 个数据文件之外，Gambit 生成一个“锁定”文件，命名为 id.lok，在这里 id 代表片段标识符。这个锁定文件的目的是要把当前 Gambit 片段的过程数据锁住以使数据文件不能被任何并行的 Gambit 片段访问和/或修改。

3.1.4 Gambit 中鼠标的使用

在 GUI 模式下，每个鼠标按键的功能根据鼠标是在菜单、表格还是在图形窗口上操作而不同。一些在图形窗口上的鼠标操作是和键盘同时进行的。

1. 在菜单和表格上的使用

Gambit 菜单和表格的鼠标操作只要求左右键而且不涉及任何键盘操作。其中大部分只需用左键操作。右键用来打开涉及命令面板上命令按钮的菜单，在一些表格上包含文本窗口，右键打开选项的隐藏菜单。

2. 在图形窗口上的使用

Gambit 图形窗口的鼠标操作有三种类型：显示操作、任务操作及点的创建。

1) 显示操作

Gambit 图形窗口显示操作既使用了鼠标的三个键，也使用了 Ctrl 键，如表 3-2 所示。

表 3-2 鼠标的显示操作

键盘/鼠标按键	鼠标运动	描 述
左键单击	拖动着指针往任一方向走	旋转模型
中键单击	拖动着指针往任一方向走	移动模型
右键单击	往垂直方向拖动指针	缩放模型
右键单击	往水平方向移动指针	使模型绕着图形窗口中心旋转
Ctrl+左键	指针对角移动	放大模型，保留模型比例。释放鼠标按键后，Gambit 放大显示
两次中键单击		在当前视角前直接显示模型

2) 任务操作

Gambit 图形窗口任务操作允许在 Gambit 表格中选中实体和执行动作中应用鼠标的三个键和 Shift 键。有两种任务操作类型：选中实体和执行动作。

(1) 选中实体

很多 Gambit 模型和网格操作要求指定一个或更多的实体操作，有两种方法来为 Gambit 操作指定一个实体。

- 在指定表格中的列表框中输入实体名字或从列表中挑选一个。
- 用鼠标直接在图形窗口模型中选中实体。

当用鼠标从显示在图形窗口中的模型中选择一个实体时，Gambit 把这个实体名字插入当前活动的列表中，就像已经在当前指定的表格中输入实体的名字一样。

Gambit 实体选中操作有两种不同的类型，都用到 Shift 键。通过指南向导，将看到像 Shift-left-click 的表达式，这说明单击鼠标左键时必须按住 Shift 键。两种选中实体的操作如表 3-3 所示。

表 3-3　鼠标的选中操作

操　作	描　述
Shift-left-click	加亮图形窗口中的实体并且包含在当前活动的列表中
Shift-middle-click	在给定类型的相邻实体间切换

为了选中一组目标，Shift-left-drag 拖出一个方框包围这些目标，方框没必要完全包围目标，只要有一部分就可以了。

(2) 执行动作

当在图形窗口中执行 Shift-left-click 操作时，Gambit 接受对一个实体的选择并把它聚焦在表格的列表中。假如当前的列表是表格中最后一个，Shift-right-click 执行的操作与当前打开的表格相联系，这种情况下，Shift-right-click 操作与单击 Apply 按钮的操作功能是一样的。

3) 点的创建

当在二、三维建模时，用鼠标操作直接在屏幕需要的地方创建点。

3.2 Gambit 建模及网格划分

3.2.1 二维建模

建立模型是划分网格的先决条件。在命令面板中单击 Geometry 按钮，进入 Geometry(几何体)面板，如图 3-6 所示，从左往右依次是划分点、线、面、体和组的按钮。

二维模型一般按照点、线、面的顺序来进行，下面分别说明。

1. 点的创建

单击 Geometry 面板中的 Vertex 按钮，进入 Vertex 对话框，如图 3-7 所示；单击 Vertex Create 按钮，在 Create Real Vertex 对话框中输入点的坐标；再单击 Apply 按钮，就可以划分点。计算出计算域的各个顶点的坐标，依次划分这些顶点。

如图 3-8 所示，在 Gambit 中点的划分方式有 7 种：根据坐标划分、在线上划分、在面上划分、在体上划分、在交点上划分、在质心上划分和在投影上划分。在建模实际中根据不同的需要来选择不同的划分方式。

Vertex 中常用的命令还有 Move/Copy、Undo、Redo 和 Del。

1) Move/Copy 命令

在如图 3-7 所示的 Vertex 面板中单击按钮，显示如图 3-9 所示的 Move/Copy Vertices 对话框。

当复制或移动一个点时，首先要选择需要作用的点。在该对话框中单击 Vertices 右边的输入栏，输入栏以高亮黄色显示，表明可以选择需要的点。

在 Gambit 中选择一个对象的方法有两种。

- 按住 Shift 键，单击选择的对象，该对象被选中，以红色显示。
- 单击输入栏右方的向上箭头，就会出现一个对话框，从对话框中可以选择需要的

点的名称，如图 3-10 所示。为了便于记忆，建议在划分对象的时候要起一个便于记住的名字。

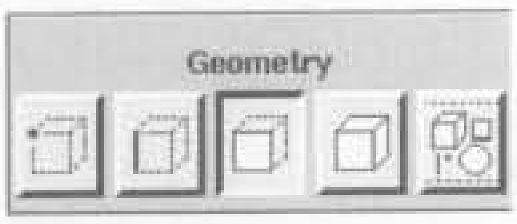


图 3-6 Geometry 面板

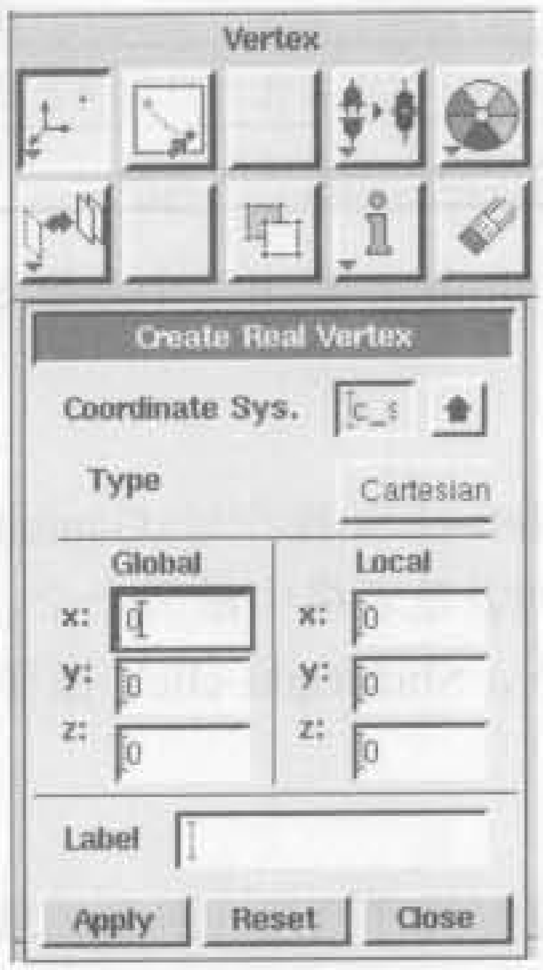


图 3-7 Vertex 面板与 Create Real Vertex 对话框

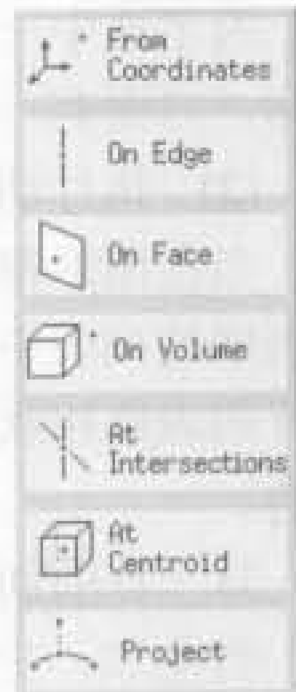


图 3-8 点的划分方式

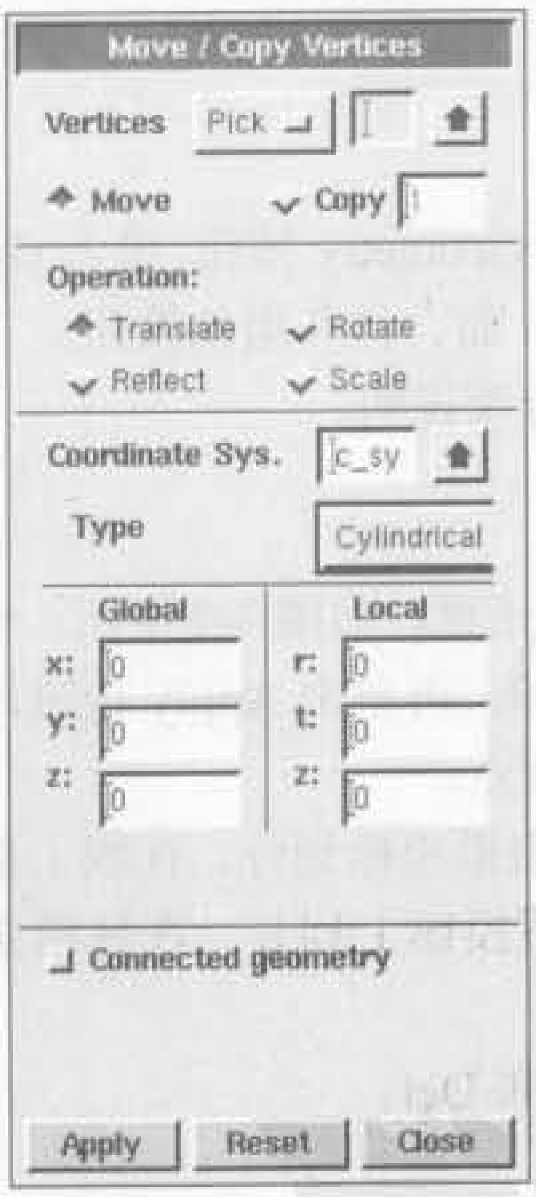


图 3-9 Move/Copy Vertices 对话框



图 3-10 Vertex List(Multiple)对话框

同时，Gambit 还提供了 3 种不同的坐标系，即直角坐标系、柱坐标系和球坐标系。在 Move/Copy Vertices 对话框的坐标类型的选项中，可以选择不同的坐标系。

2) Undo 和 Redo 命令

Undo 命令用以消除上一步操作的内容，Redo 命令用来恢复上步取消的操作。按钮图标为Undo、Redo。

3) Del 命令

Del 命令用来删除一些错误操作或不需要的对象。单击 Del 按钮，在视图中选择需要删除的对象，再单击 Apply 按钮即可。

2. 线的创建

在 Geometry 面板中单击 Edge 按钮，就可以进行线的创建和编辑，如图 3-11 所示。

在 Gambit 中，最常用的是直线的创建。在 Edge 面板中单击 Create Straight Edge 按钮，在视图中选择需要连成线的点，单击 Apply 按钮即可。

除了划分直线外，Gambit 还可以划分其他的一些线段，如圆弧、圆、倒角、椭圆等，如图 3-12 所示。

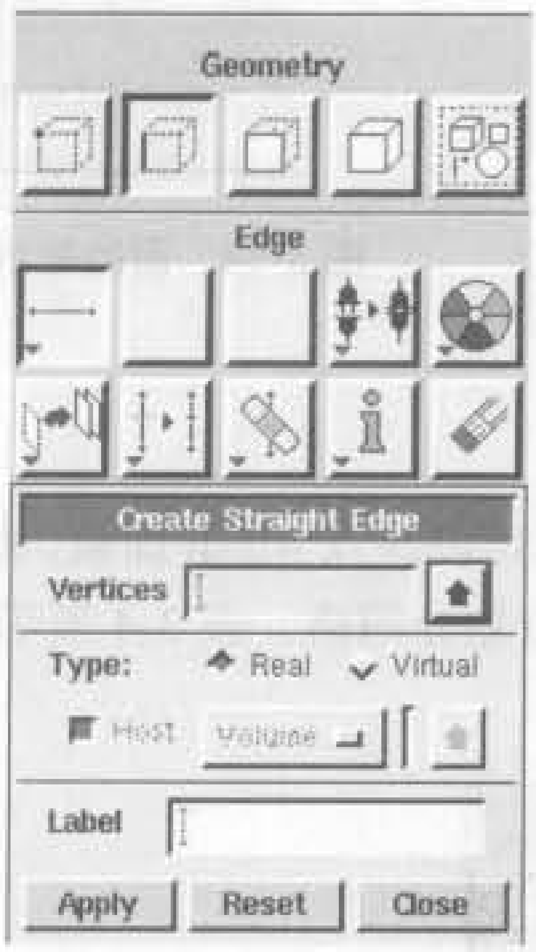


图 3-11　Edge 面板和 Create Straight Edge 对话框



图 3-12　线划分方式

Edge 面板中常用的还有合并、分离等命令按钮，即可以把两条线段合成一条，也可以将一条线段分成两条，这些可以为面的划分和网格划分提供方便。因为面的划分需要一个封闭的曲面。

3. 面的创建

在 Geometry 面板中单击 Face 按钮，就可以进行面的创建和编辑，如图 3-13 所示。面的创建比较简单，只需选择组成该面的线，单击 Apply 按钮即可，需要注意的是这些线必须是封闭的。

在面的创建中，有一个布尔运算的操作，可以使我们创建不规则形状的面。布尔运算包括三种方式：加、减、交，如图 3-14 所示。



图 3-13 Face 面板和 Creater Face from Wireframe 对话框

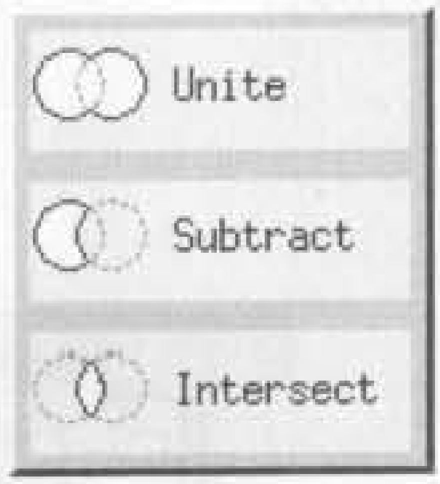


图 3-14 面的布尔运算方式

3.2.2 二维网格划分

在如图 3-3 所示的命令面板中单击 Mesh 按钮，就可以进入 Mesh 面板，如图 3-15 所示。在 Gambit 中，可以分别针对边界层、边、面、体和组划分网格。表 3-4 所示的 5 个按钮分别对应着这 5 个命令。

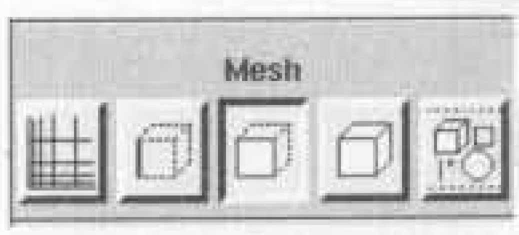


图 3-15 Mesh 面板

表 3-4 网格划分类型

	Boundary Layer(边界层)
	Edge(边)
	Face(面)
	Volume(体)
	Group(组)

1. 边界层网格的划分

在 Mesh 面板中单击按钮，即可进入 Create Boundory Layer 对话框，如图 3-16 所示。边界层网格的划分需要输入 4 组参数，分别是第一个网格点距边界的距离(First row)、网格的比例因子(Growth factor)、边界层网格点数(Rows)(垂直边界方向)以及边界层厚度

(Depth)。这 4 个参数中只要任意输入 3 组参数值即可划分边界层网格。

同时，可以选择边界层网格划分的形式。在该对话框的 Transition pattern 选项组中，系统提供了 4 种划分方式，如图 3-17 所示。

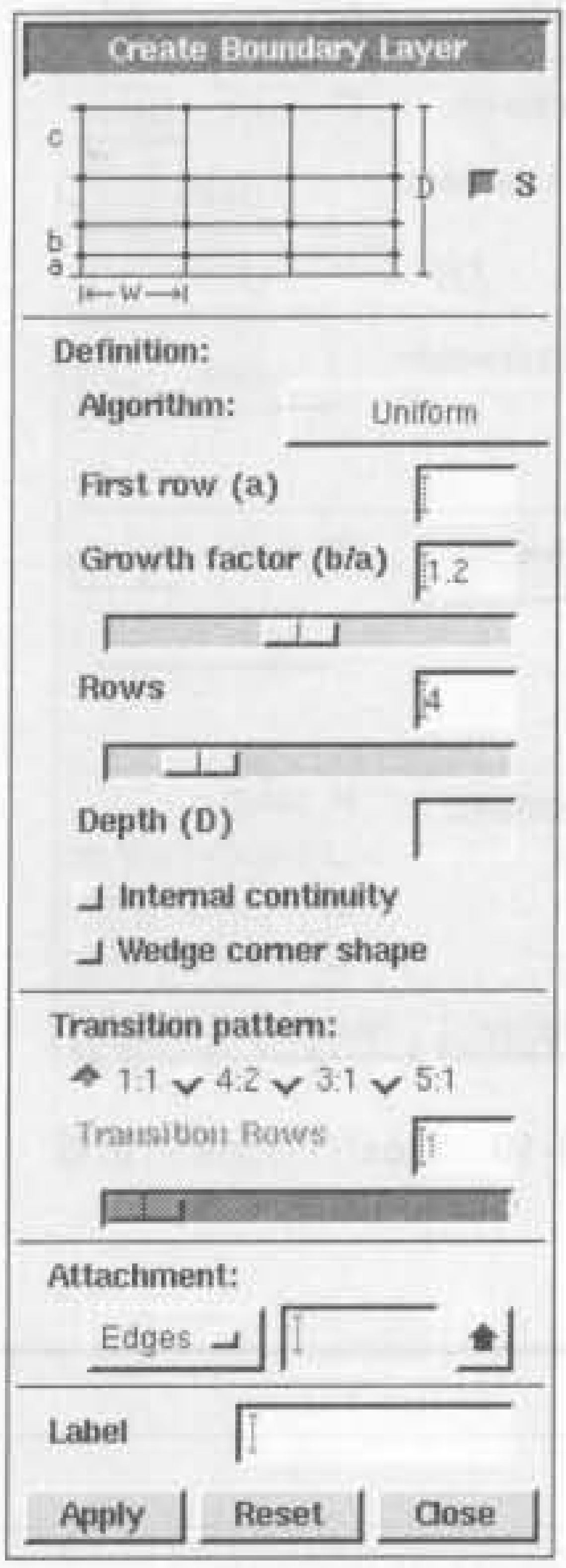


图 3-16　Create Boundary Layer 对话框

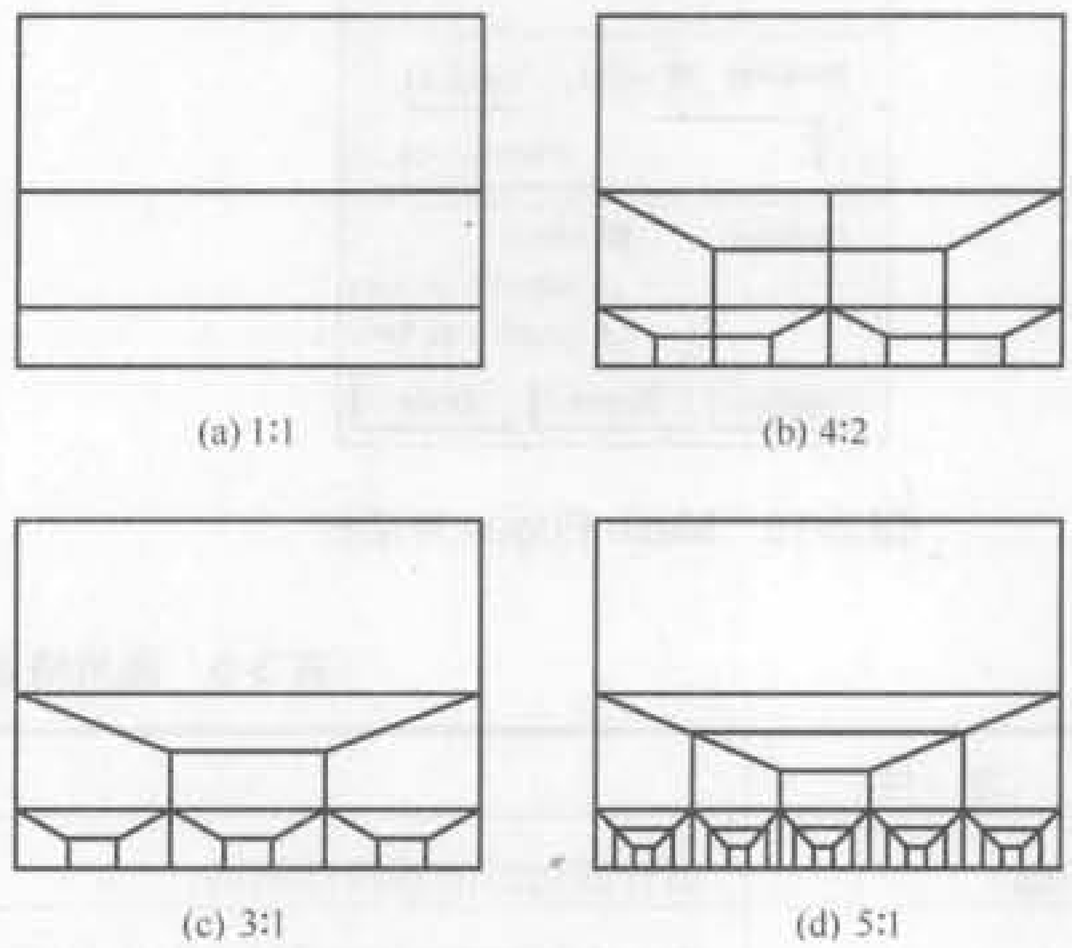


图 3-17　边界层划分方式

2. 边的划分

当划分的网格需要在局部加密或者划分不均匀网格时，首先要定义边上的网格点的数目和分布情况。

边上的网格点的分布可分为两种情况，一种是单调递增或单调递减；另一种是中间密(疏)两边疏(密)。下面结合实例介绍边上网格点的划分。

单击 Mesh 面板中的□按钮，进入 Mesh Edges 对话框，如图 3-18 所示。

3. 面的网格划分

单击 Mesh 面板中的□按钮，进入 Mesh Face 对话框，如图 3-19 所示。

Gambit 对于二维面的网格划分提供了 3 种网格类型：四边形、三角形和四边形/三角形混合，同时还提供了 5 种网格划分的方法。表 3-5、表 3-6 分别列举了 5 种网格划分的方法及其适用类型。

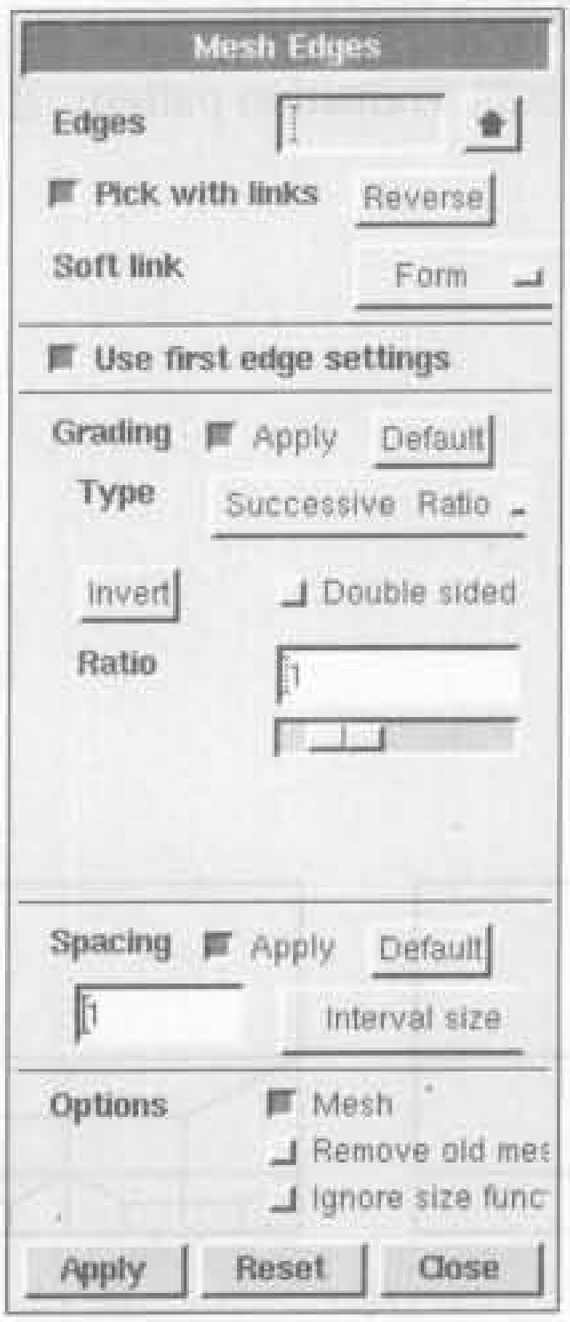


图 3-18 Mesh Edges 对话框

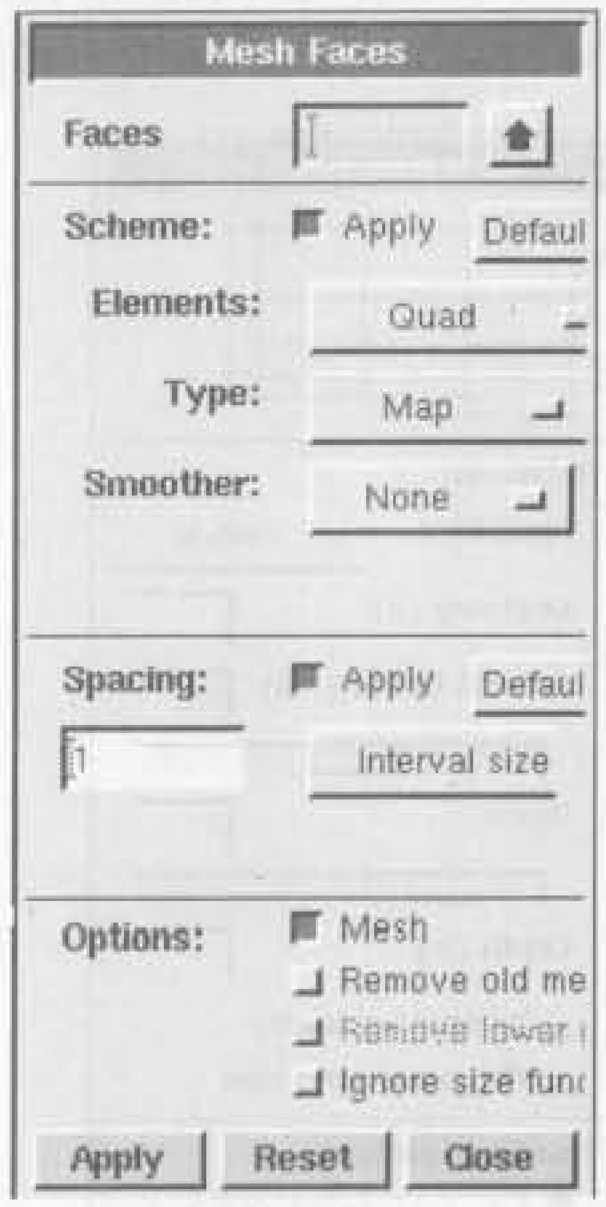


图 3-19 Mesh Faces 对话框

表 3-5 面网格划分方法

方 法	描 述
Map	划分四边形的结构性网格
Submap	将一个不规则的区域划分为几个规则区域并分别划分结构性网格
Pave	划分非结构性网格
Tri Primitive	将一个三角形区域划分为三个四边形区域并划分规则网格
Wedge Primitive	在一个楔形的尖端划分三角形网格，沿着楔形向外辐射，划分四边形网格

表 3-6 划分方法对应的网格类型

方 法	适用类型		
	Quad	Tri	Quad/Tri
Map	X		X
Submap	X		
Pave	X	X	X
Tri Primitive	X		
Wedge Primitive			X

4. 边界的定义

在 Gambit 中，可以先定义好各个边界条件的类型，具体的边界条件取值在 Fluent 中确定。

(1) 在菜单栏中选择 Fluent 4 或 Fluent 5/6。这个步骤是不可缺少的，它相当于给 Gambit 定义了一个环境变量。设置完之后，定义的边界条件类型和 Fluent 5/6 中的边界类型相对应。

(2) 在命令面板中单击按钮，进入 Specify Continuum Types 对话框，如图 3-20 所示。

(3) 单击按钮，出现 Specify Boundary Types 对话框，如图 3-21 所示。

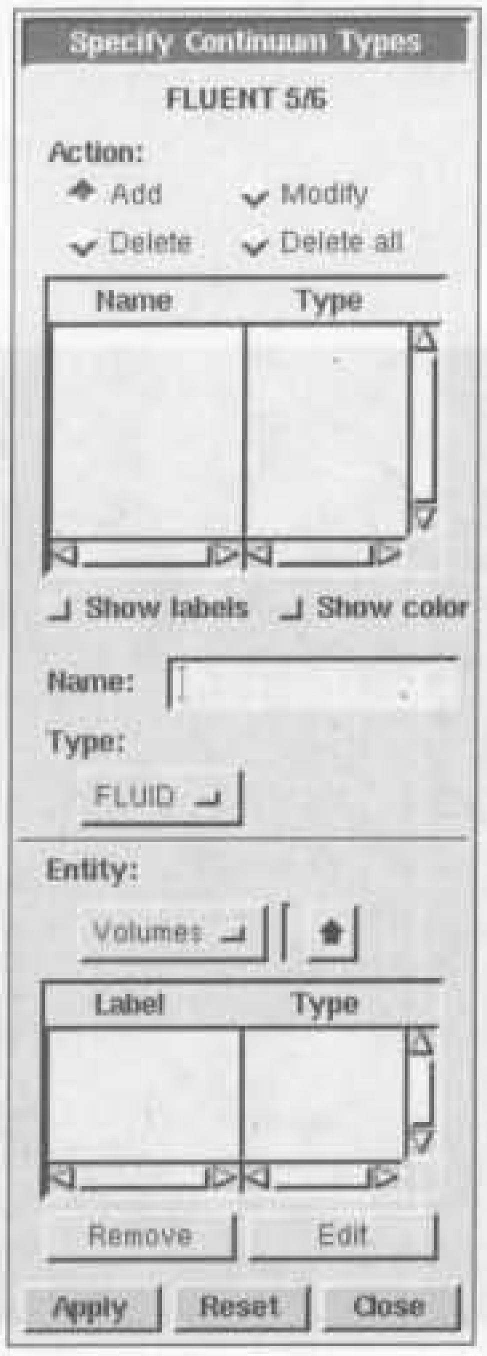


图 3-20　Specify Continuum Types 对话框

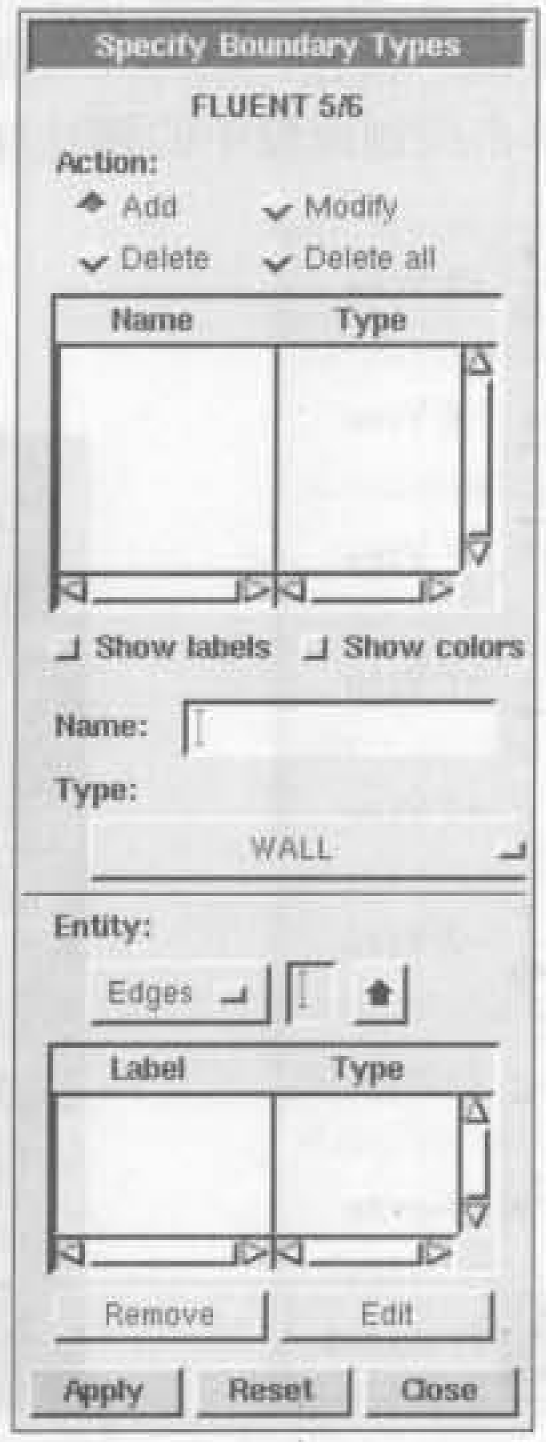


图 3-21　Specify Boundary Types 对话框

3.2.3　三维建模

相对于二维建模而言，三维建模与二维建模的思路有着较大的区别。二维建模主要遵循点、线、面的原则，而三维建模则更像搭积木一样，由不同的三维基本造型组合而成，因此在建模的过程中更多地用到了布尔运算及 Pro/ENGINEER 等其他建模工具。

1. 三视图的使用

在建立三维图形的时候，使用三视图有利于更好地理解图形。

如图 3-2 所示的 Gambit 的视图控制面板，在 4 个视图都是激活的情况下(在 Active 栏中，所有视图框显示为红色)，这时视图控制面板中的 10 个命令将同时作用于 4 个视图。

在划分三维图形之前，要做的第一项工作就是将 Gambit 的 4 个视图设置为顶视图、前

视图、左视图和透视图。

(1) 单击 Active 右边的后 3 个视图，取消对它们的激活，激活取消后呈灰色，如图 3-22 所示。

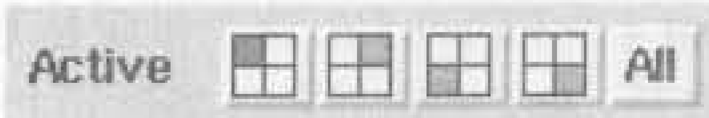


图 3-22 取消视图激活

(2) 右击视图控制面板中的坐标按钮，弹出一组坐标系，如图 3-23 所示。

(3) 单击按钮，则左上视图变成顶视图。类似地设置其他视图，如图 3-24 所示。

(4) 单击控制面板中的按钮，也可将视图设成三视图。

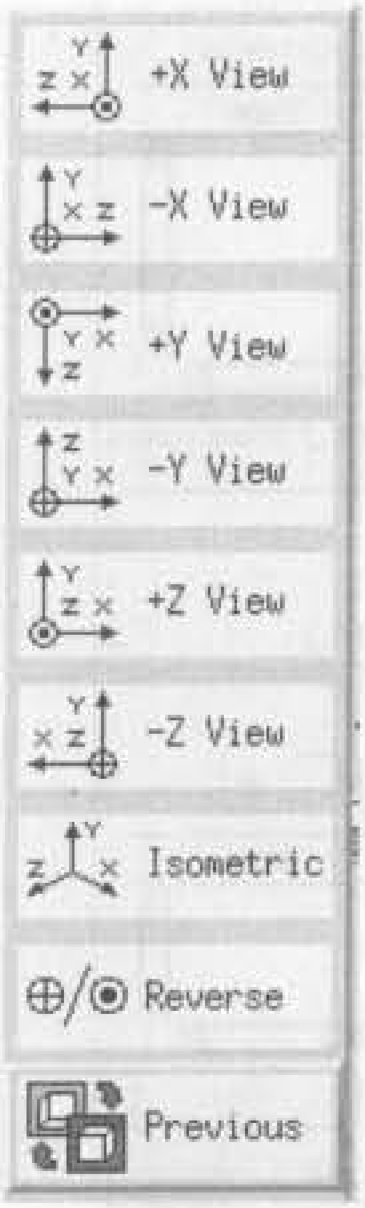


图 3-23 坐标系

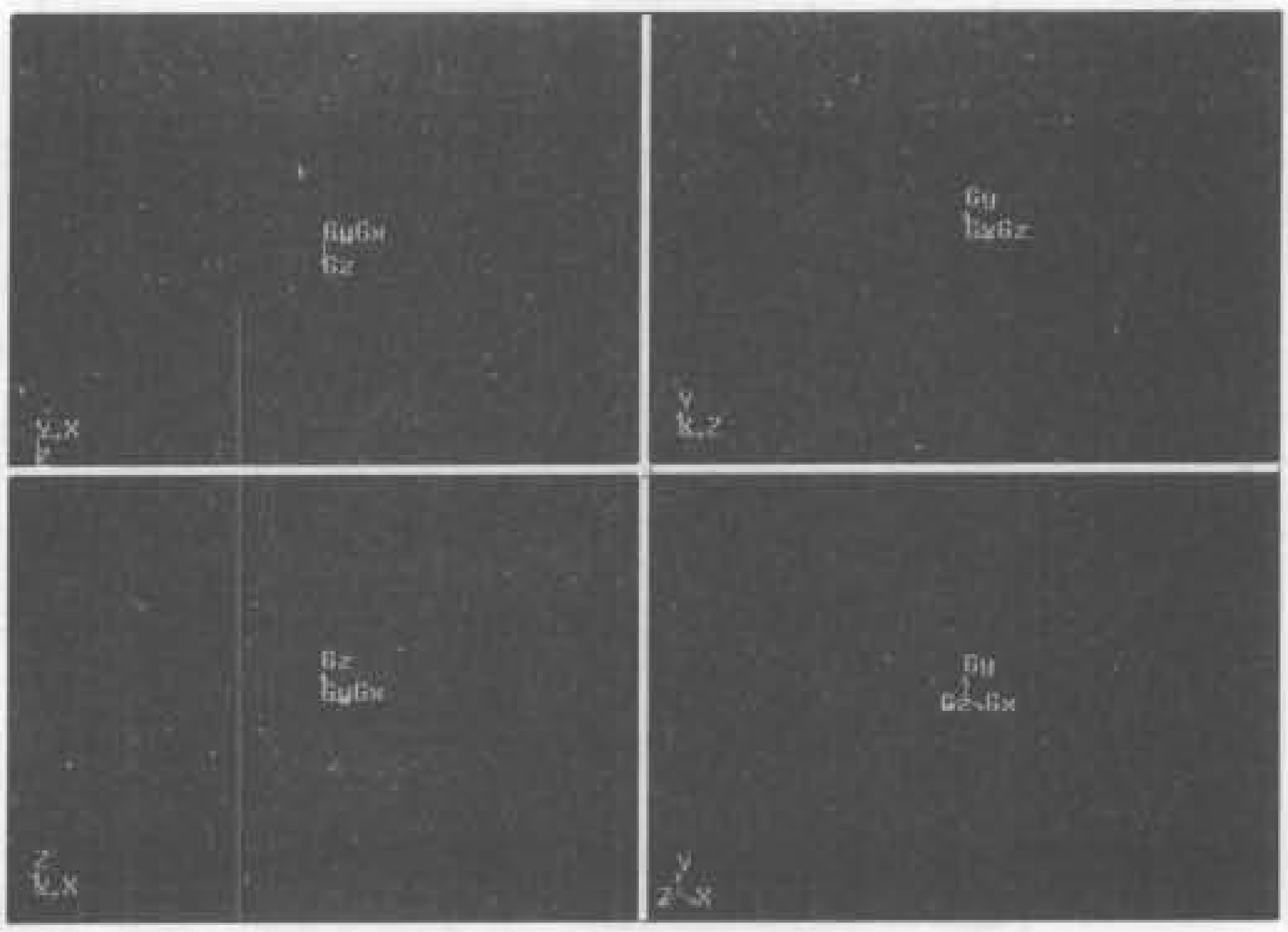


图 3-24 视图的设置

2. 基本三维模型的建立

在 Gambit 控制面板中单击按钮，显示如图 3-25 所示的 Volume(三维模型创建)面板。在 Volume 面板中右击按钮，弹出一组几何体类型，如图 3-26 所示，表示 Gambit 所能划分的基本三维几何体，主要有长方体、圆柱体、棱柱等。

3. 布尔运算的基本概念

典型的布尔运算包括并、交、减，具体功能如下。

- 并：将两个物体并成一个物体(两个物体的并集)。
- 交：两个物体的交集。
- 减：A 物体减去 B 物体。

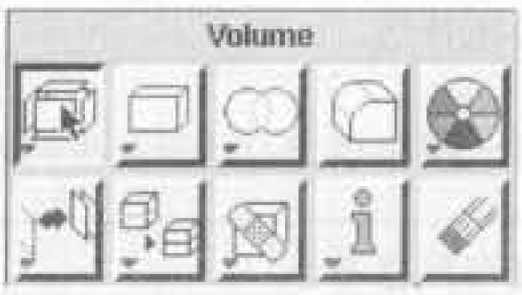


图 3-25 Volume 面板



图 3-26 基本三维几何体类型

3.2.4 三维网格划分

三维几何体网格的划分与二维的基本一样，但三维物体的网格划分比较难以把握，尤其是对局部的加密。

1. 实体(三维几何体)的划分命令

单击 Mesh 面板中的 Volume 按钮，出现如图 3-27 所示的 Volume 面板和 Mesh Volumes 对话框，表 3-7 给出各命令的功能。

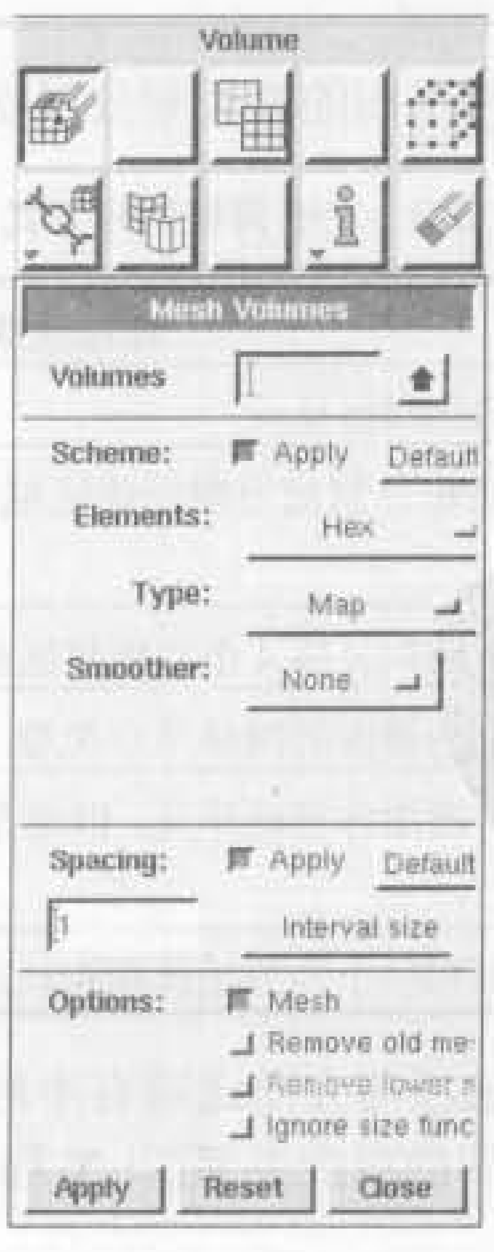


图 3-27 Volume 面板和 Mesh Volumes 对话框

表 3-7 实体网格划分命令功能

图 标	命 令	详细说明
	Mesh Volumes	为整个体积生成网格节点
	Smooth Volume Meshes	调整体积网格节点位置，提高节点步长的一致性
	Set Volume Element Type	指定用于整个模型的体积单元类型
	Link Volume Meshes · Unlink Volume Meshes	建立或者断开体积之间的坚固连接
	Modify Meshed Geometry	将网格边转化为拓扑边
	Summarize Volume Mesh Check Volume Meshes	在图形窗口中显示网格信息，显示三维网格质量信息
	Delete Volume Meshes	从体积中删除现有网格节点

2. 三维网格的类型与方式

Gambit 允许用户采用如表 3-8 所示的单元(Elements)类型进行三维网格划分。

表 3-8 三维网格单元类型

选 项	详细说明
Hex	指定网格仅仅包含六面体网格单元
Hex/Wedge	指定网格主要由六面体网格单元组成，但是也包括在适当的位置的楔形网格
Tet/Hybrid	指定网格主要由四面体网格构成，但是在适当的位置可以包含六面体、锥形和楔形网格单元

对于不同的单元类型，可以采用不同的方式划分网格，如表 3-9 所示。

表 3-9 三维网格划分方式

选 项	详细说明
Map	生成一般六面体结构化网格单元
Submap	将一个不可图示化体积分割成可图示化区域并在每个区域生成六面体结构化网格单元
Tet Primitive	将一个含 4 个侧面的体积分成 4 个六面体区域并在每个区域生成可图示化网格
Cooper	扫描整个体积的指定的源面的网格节点类型
Tet/Hybrid	指定该网格主要包含四面体网格单元，但是在合适的位置也可以包含六面体、锥体和楔形单元
Stairstep	生成普通六面体网格和一个与原始体积形状近似的平滑的体积

如上所述，三维网格划分类型选项与方法选项有不同的组合。表 3-10 中显示了每个实体网格划分类型选项和方式选项之间的对应关系(以 X 标记的单元代表选项的允许组合)。

表 3-10　三维网格的 Elements 和 Type 的组合

Type 选项	Elements 选项		
	Hex	Hex/Wedge	Tet/Hybrid
Map	X		
Submap	X		
Tet Primitive	X		
Cooper	X	X	
TGrid			X
Stairstep	X		

表 3-10 中显示的每个允许组合都构成了对于任意给定体积的一种独特的网格节点形式。另外，每个组合与相关的限制(一组控制其应用的体积类型)相关联。下面详细介绍表 3-10 中列举的可能的选项组合的形式和相关的限制。

(1) 表 3-10 中列举的 Type 选项中，仅仅 Cooper 选项与多个 Elements 选项相关。

(2) 当用户在 Mesh Volumes 对话框中指定一个实体时，Gambit 将自动评估该体积关于形状、拓扑结构特点以及顶点类型，并设置 Scheme 选项按钮来反映一种要求的实体网格划分方式。如果为一个网格划分操作指定了多个实体，则 Scheme 选项按钮提供的方式将反映最近选定的实体要求的方式。如果用户通过 Mesh Volumes 对话框中的 Scheme 选项按钮强制一种网格划分方式，Gambit 将指定的方式应用于所有当前选定的实体。

(3) 表 3-10 中列举的一些网格划分方式生成的网格节点类型不能被主菜单栏中的 Gambit Solvers 菜单包含的一些解算器使用。表 3-11 中显示了 Solvers 菜单中的可用解算器与上面列举的网格划分类型方式之间的对应关系(注意：Fluent 4 解算器要求一种结构化网格，NEKTON 解算器要求六面体网格单元)。

表 3-11　网格类型方式与解算器的匹配

解算器	Type 选项					
	Map	Submap	Tet Primitive	Cooper	TGrid	Stairstep
FIDAP	X	X	X	X	X	X
Fluent/UNS	X	X	X	X	X	X
Fluent 5	X	X	X	X	X	X
Fluent 4	X	X		X		X
NEKTON	X	X	X	X		X
RAMPANT	X	X	X	X	X	X
Polyflow	X	X	X	X	X	X
Generic	X	X	X	X	X	X

3.2.5　与 CAD 软件的接口

Gambit 只适用于创建简单的三维几何体，对于复杂实体的建模有一定的难度，这时

Gambit 允许从主流的 CAD/CAE 系统如 Pro/E、UGII、I-DEAS、CATIA、SolidWorks、Ansys、Patran 导入几何体和网格。图 3-28 是 Gambit 能识别的文件格式。

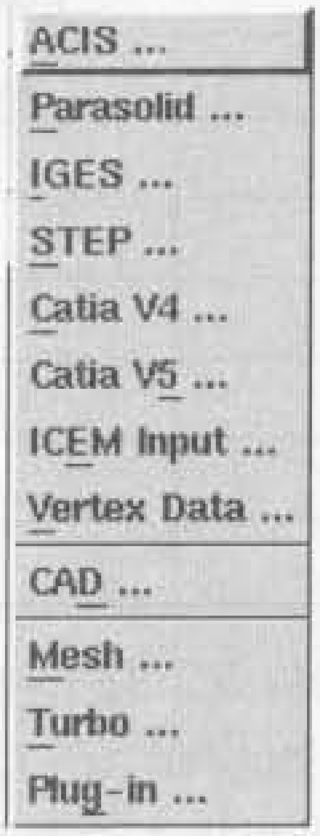


图 3-28 Gambit 能识别的文件格式

3.3 建模及网格划分实例

3.3.1 二维轴对称维多辛斯基曲线喷嘴

图 3-29 所示为维多辛斯基曲线喷嘴示意图，图中的维多辛斯基曲线虽然在 Gambit 中也能划分，但曲线的光滑效果不如 CAD 中的好。因此在遇到复杂几何体时，可以考虑在 CAD 中绘制部分图形然后在 Gambit 中进行组装。

1. 在 AutoCAD 中划分维多辛斯基曲线

(1) 利用 pline 命令将维多辛斯基曲线上的各点坐标连成一条折线。

(2) 利用 pedit 命令使折线光滑。

(3) 划分其他轮廓线，如图 3-30 所示。

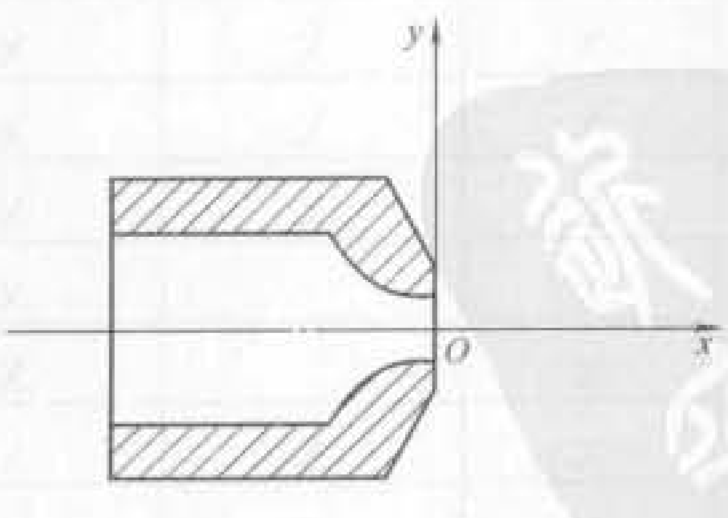


图 3-29 维多辛斯基曲线喷嘴示意图

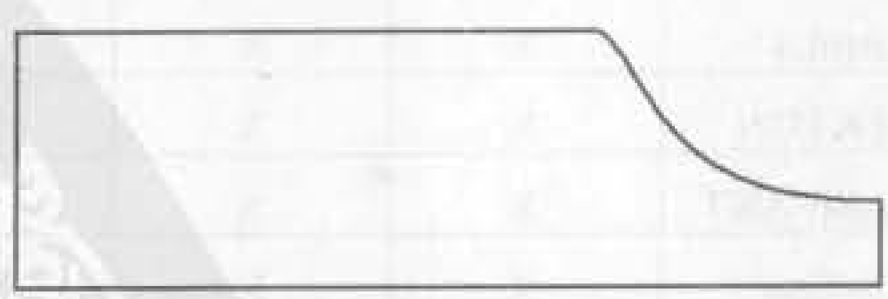


图 3-30 CAD 中划分的维多辛斯基轮廓线

2. 输出为 ACIS 的.sat 文件

二维图形要输出为.sat 文件，必须是一个 region 图形。

(1) 输入 region 命令，或在命令面板中单击按钮。

(2) 选择喷嘴轮廓线，右击或按 Enter 键。

(3) 选择 File→Export 命令，选择保存类型为 ACIS(*.sat)，输入文件名为 jet.sat。

(4) 选择喷嘴轮廓线，右击或按 Enter 键。

3. 在 Gambit 中输入.sat 文件

在 Gambit 中选择 File→Import→ACIS 命令，输入文件名，如图 3-31 所示，单击 Accept 按钮，即可将 CAD 中划分的图形读入 Gambit，如图 3-32 所示。

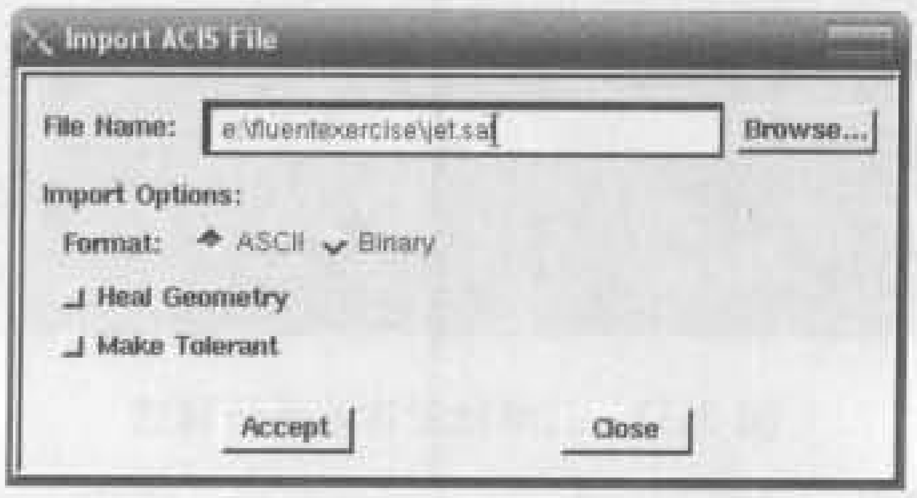


图 3-31　导入文件对话框

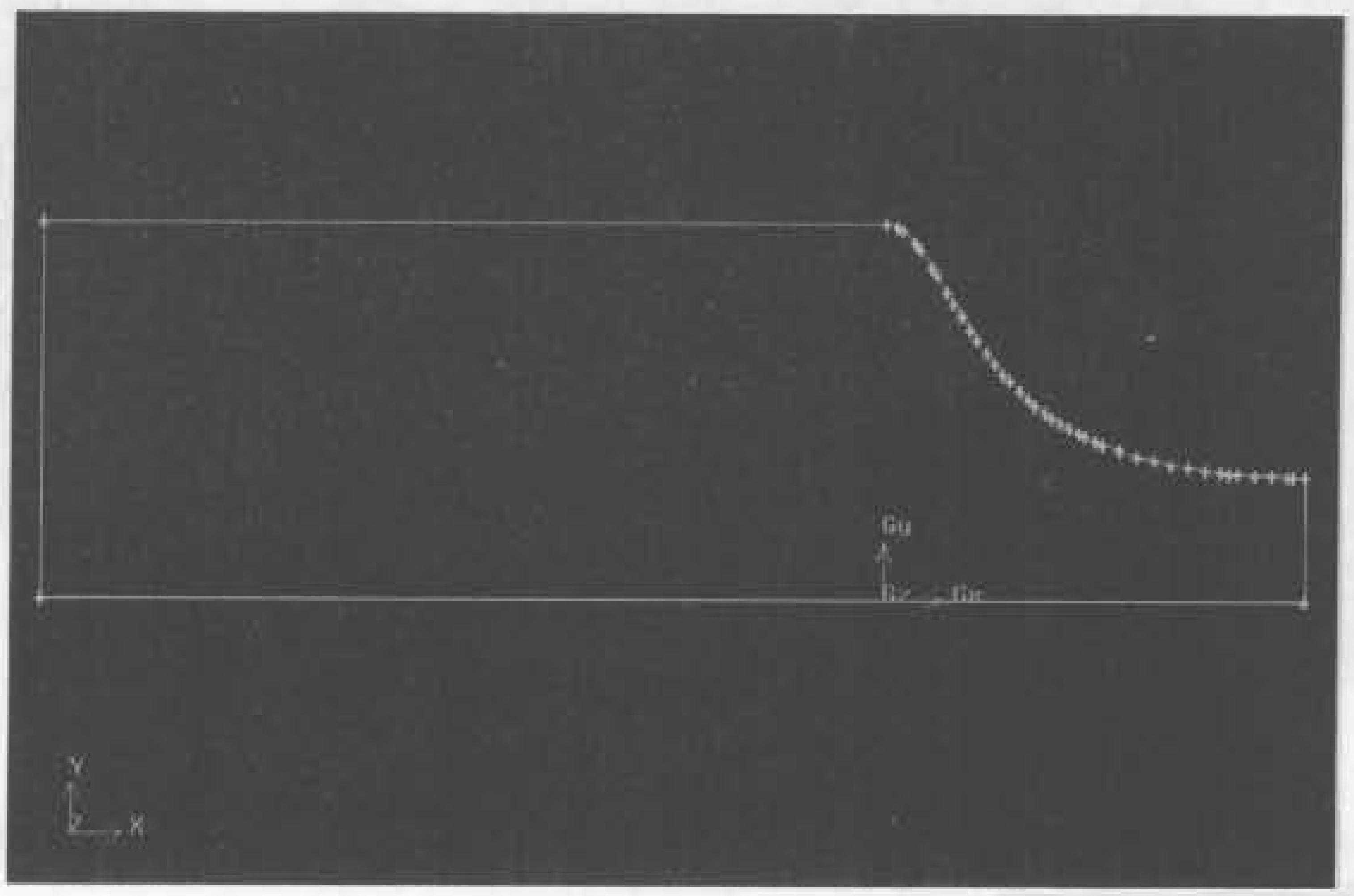


图 3-32　输入图形

注意：由于 Gambit 中只能利用坐标参数进行定位，所以在 CAD 中划分图形时要注意选好坐标(如起始点为原点坐标)。

4. 完成模型的其他部分

如图 3-33 所示，完成模型的其他部分，将喷嘴的外流场组成一个面，计算域为 20D×5D。

5. 划分网格

(1) 喷嘴内部的面(face1)，定义网格数为 80×50，网格类型为四边形 map 网格，如图 3-34 所示。

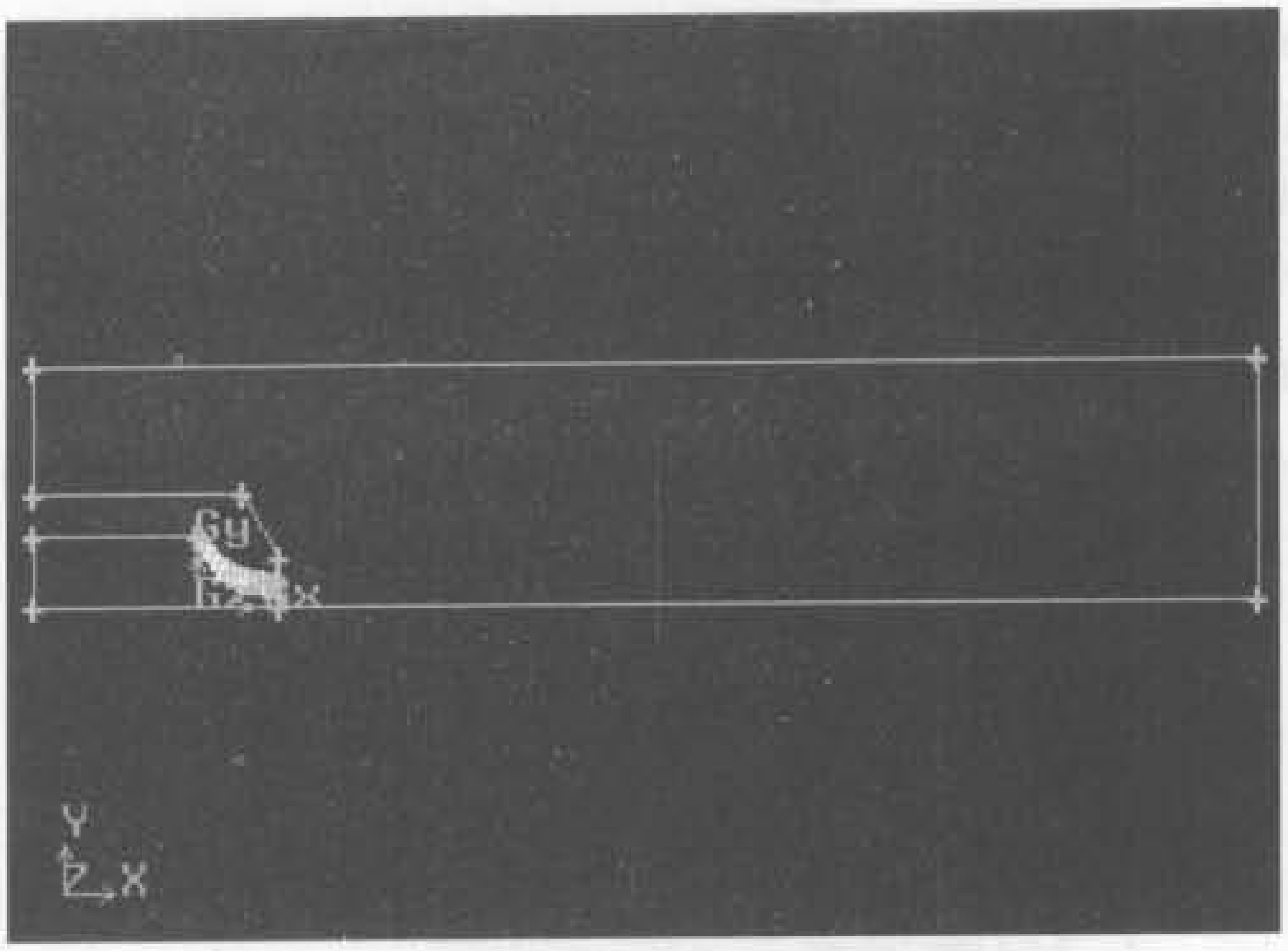


图 3-33 二维轴对称喷嘴计算域

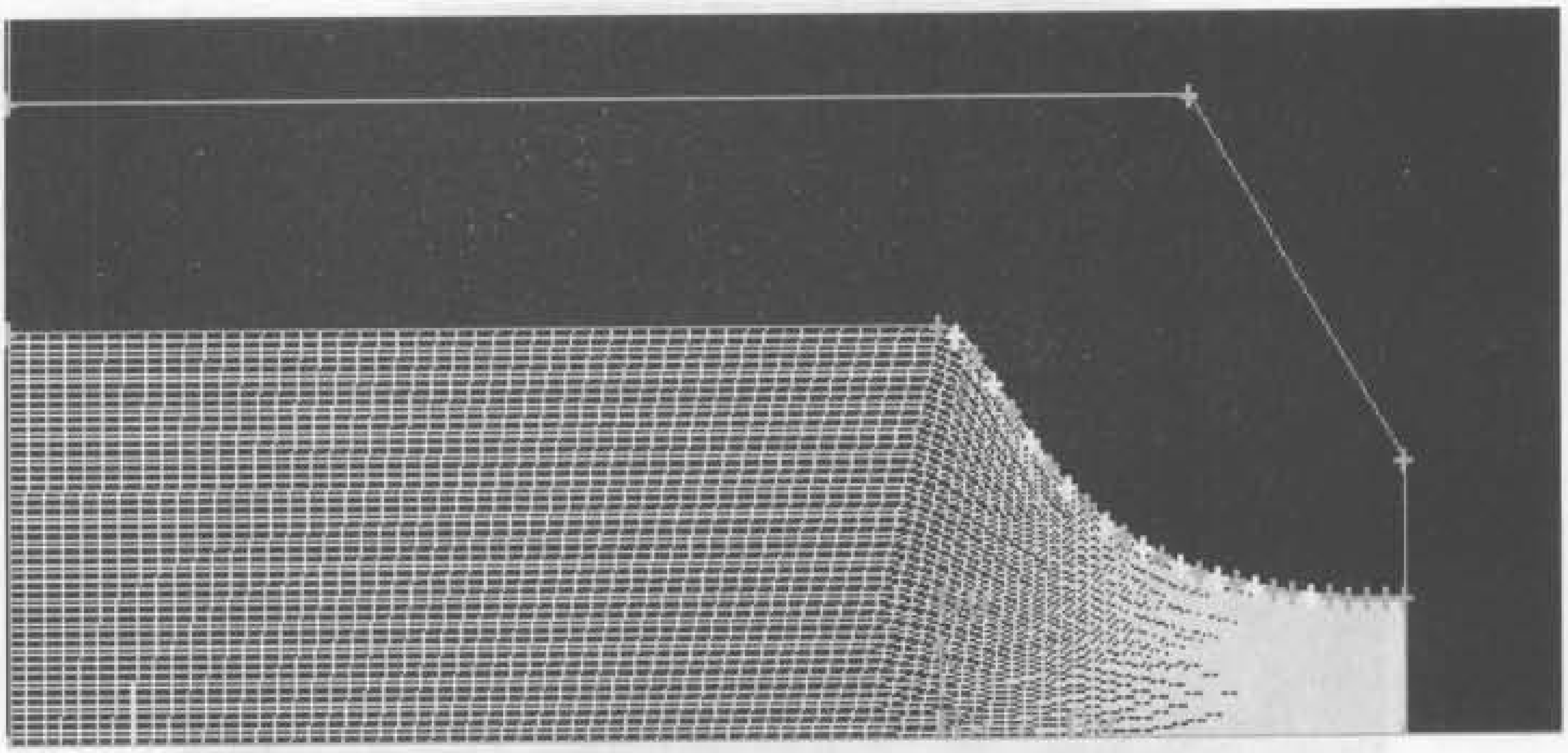


图 3-34 face1 的划分

(2) 喷嘴外部的面(face2)，定义轴线上网格点为 240 个。定义喷嘴外轮廓线的网格点数，如图 3-35 所示。

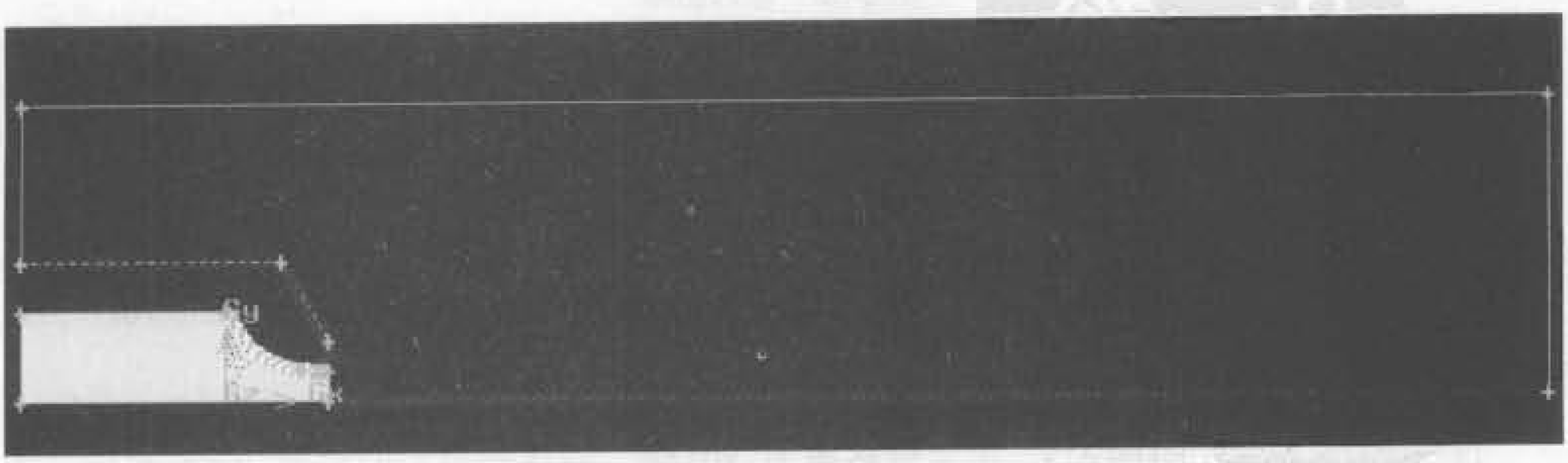


图 3-35 face2 边线的网格点

注意： 对于网格的划分，如果要求控制网格的密度，可以遵循从线到面的原则，但是对于多边形区域而言，不能将所有边的网格点都定死，必须有一些边不定义网格。如四边形区域，一般只定义相邻两个边的网格。至于多边形区域怎样定义边上的网格，必须在实践中不断地尝试。

(3) 划分外区域的网格，网格类型为 submap，如图 3-36 所示。

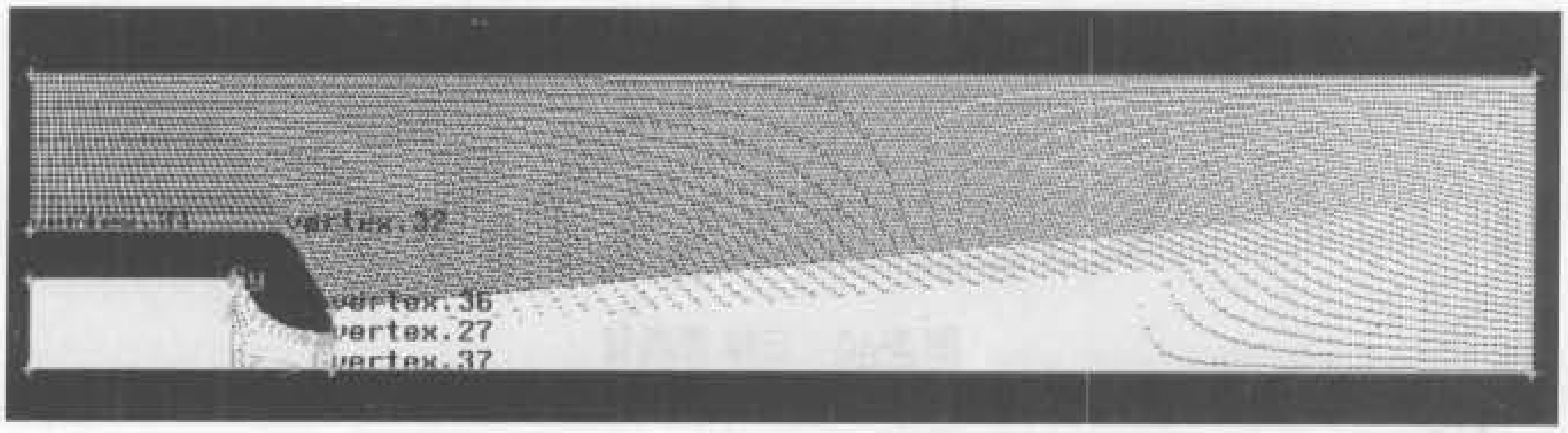


图 3-36　外区域网格的划分

6. 定义边界条件

(1) 选择 Solver→Fluent 5/6 命令。

(2) 单击按钮。

(3) 定义各边界条件。

(4) 单击按钮，将两个面设为同一个连续体(fluid)。

注意： 对于一个复杂的几何体而言，在网格划分时必定要划分为多个区域，将这些区域定义到一个统一的连续体中，这样，不同区域间的分隔线就会被默认为内部网格点。

7. 将网格输出为.msh 文件

在 Gambit 中选择 File→Export→Mesh，输入文件名，选中 Export 2-D(X-Y)Mesh 复选框，单击 Accept 按钮，如图 3-37 所示，完成网格文件输出。

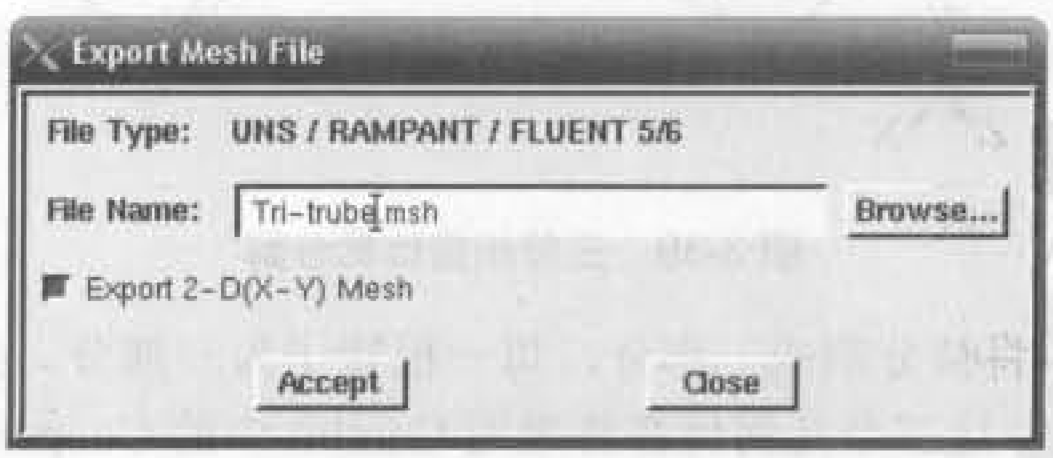


图 3-37　Export Mesh File 对话框

3.3.2　三维贯通管

如图 3-38 所示，几何体是由三根相贯的管子组成，每根管子的直径为 6，长度为 10。三根管子相互正交。此几何体可以由三根相贯的圆柱体和一个位于相交角处的 1/8 球体

组成。

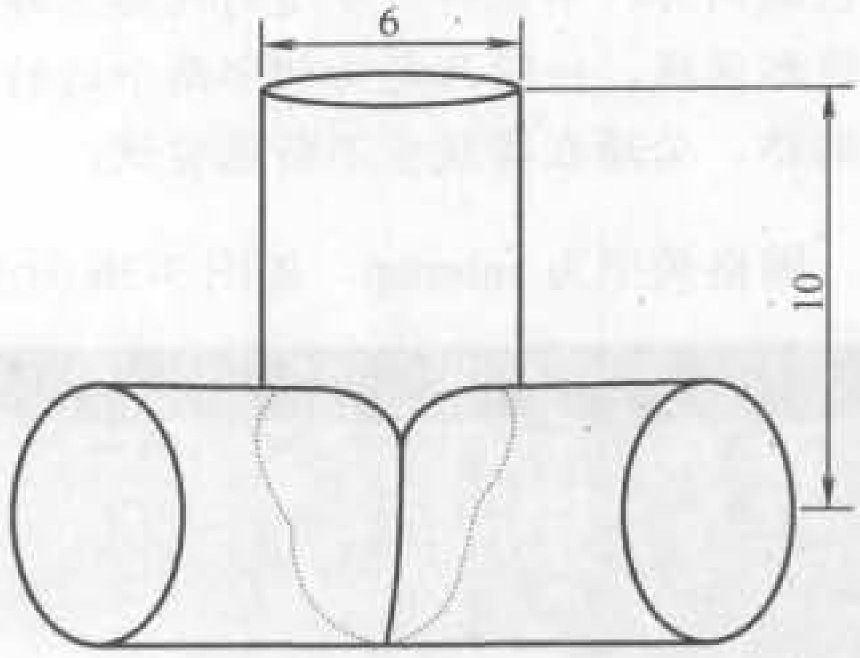


图 3-38 三维贯通管

1. 分析

(1) 用一个方体对三管相贯体的一部分进行分割，产生的几何体就是一个 1/8 球体，位于三管相贯的角落，如图 3-39 所示。这个形体，形状上与一个四面体非常近似，因此，可以用 Gambit 中的 Tet Primitive 方法进行网格划分。注意，生成的是四面体拓扑的六面体网格，不是四面体网格。

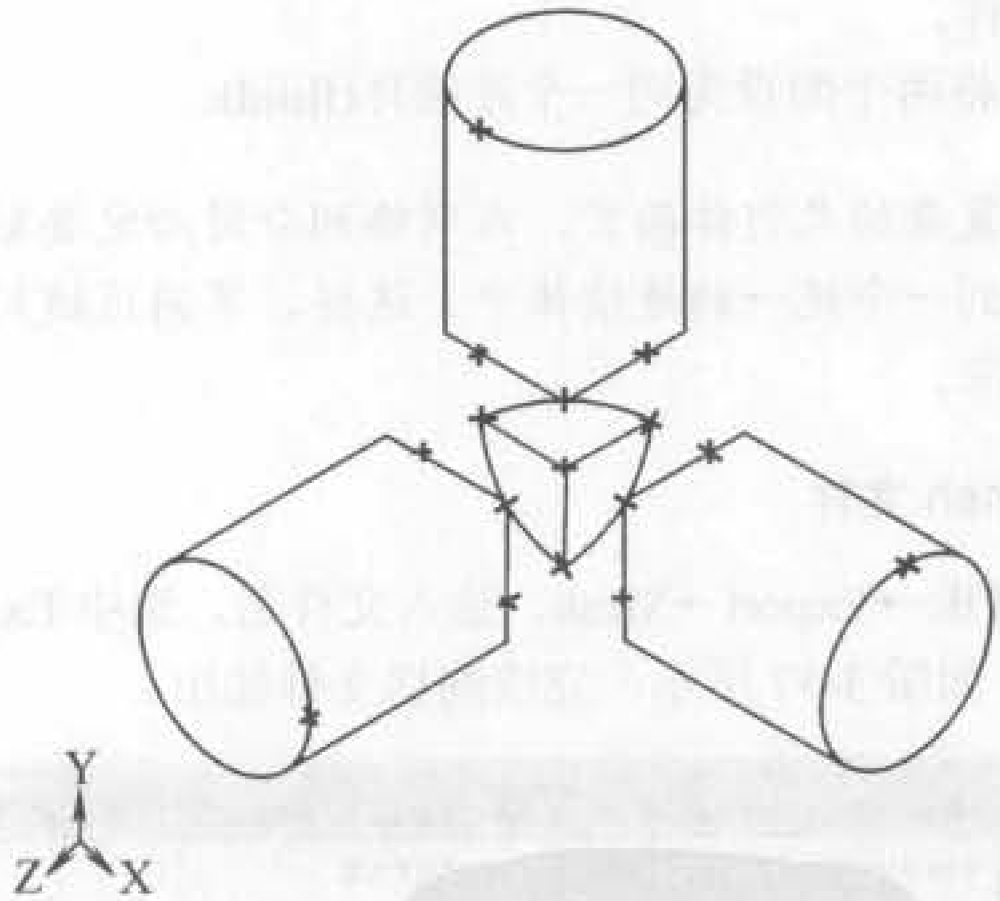


图 3-39 三管相贯体的分解

(2) 剩下的几何体将被分割成三部分，每一根管子为一部分。为了做到这一点，必须先创建一条边和三个面。这三个几何体都将采用 Gambit 中的 Cooper 方法来进行网格划分。

2. 建模步骤

1) 创建三管相贯几何体

选择 Geometry→Volume→Create Volume (Cylinder)命令，打开如图 3-40 所示的 Create Real Cylinder 对话框。

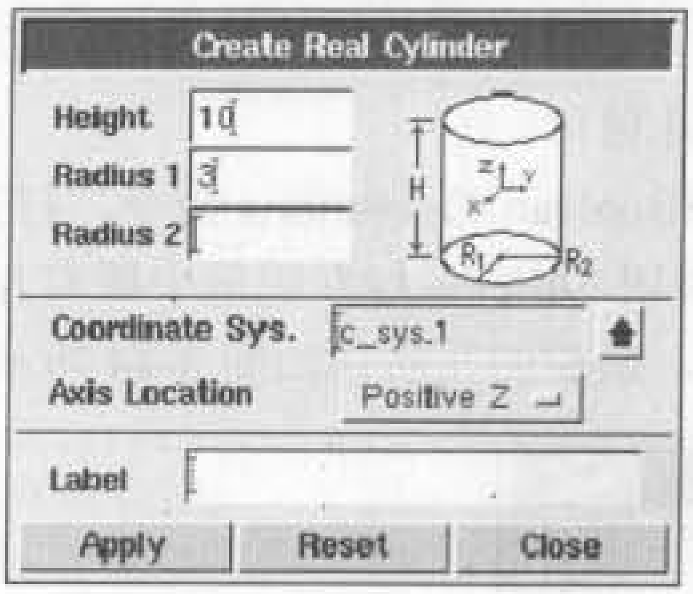


图 3-40　Create Real Cylinder 对话框

(1) 创建第一根管子

① 在 Height 文本框中输入 10。

② 在 Radius1 文本框中输入 3；Radius 2 文本框可以不输值，Gambit 将默认把其值设为 Radius1 的值。

③ 在 Axis Location 右边列表中选择 Positive Z(默认值)。

④ 单击 Apply 按钮。

(2) 创建第二根管子

第①、②步和上面一样，在 Axis Location 右边列表中选择 Positive X。

(3) 创建第三根管子

同样地，只是在 Axis Location 右边列表中选择 Positive Y。

2) 观察三柱体

单击 Global Control 工具箱中的 Fit to Window 命令按钮，观察三个柱体。可以通过按住鼠标左键并拖动来旋转视图，柱体视图如图 3-41 所示。

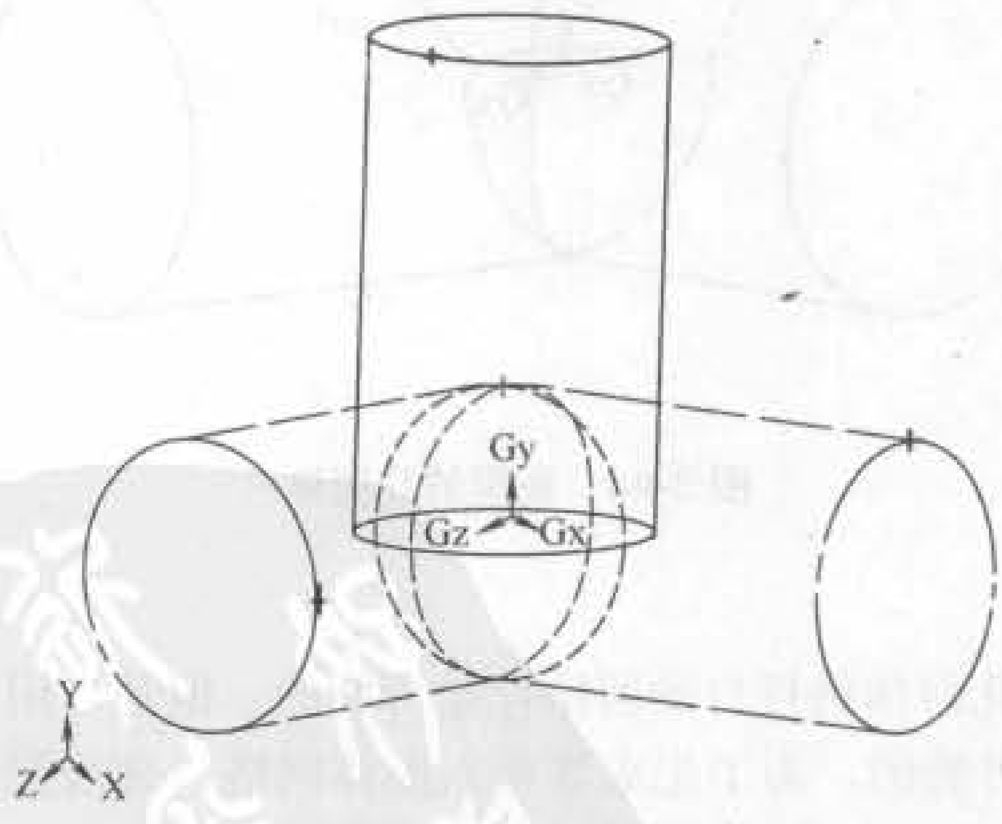


图 3-41　三管相贯体的三个柱体

3) 创建一个球体，完成基本几何体的创建

选择 Geometry→Volume→Create Volume (Sphere)命令，打开如图 3-42 所示的 Create Real Sphere 对话框。

(1) 在 Radius 文本框中输入 3。

(2) 单击 Apply 按钮。

4) 整合四个形体为一个几何体

选择 Geometry→Volume→Boolean Operations 命令，打开如图 3-43 所示的 Unite Real Volumes 对话框。在图形窗口中用 Shift+鼠标左键选中所有几何体，再单击 Apply 按钮。所有的形体被整合成一个几何体，最后的几何体如图 3-44 所示。

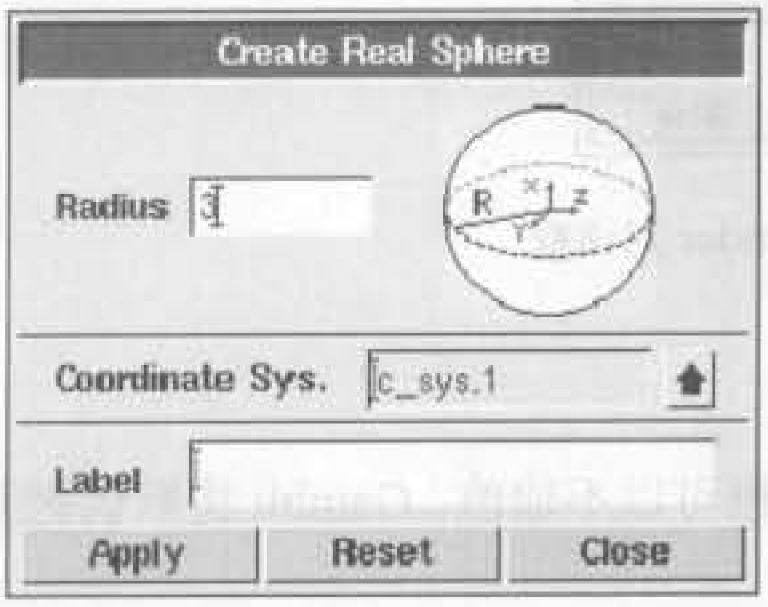


图 3-42 Create Real Sphere 对话框

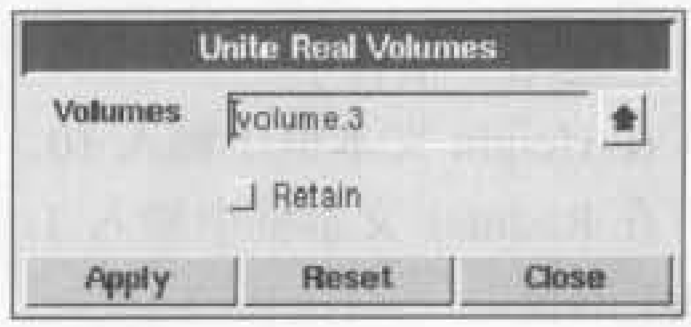


图 3-43 Unite Real Volumes 对话框

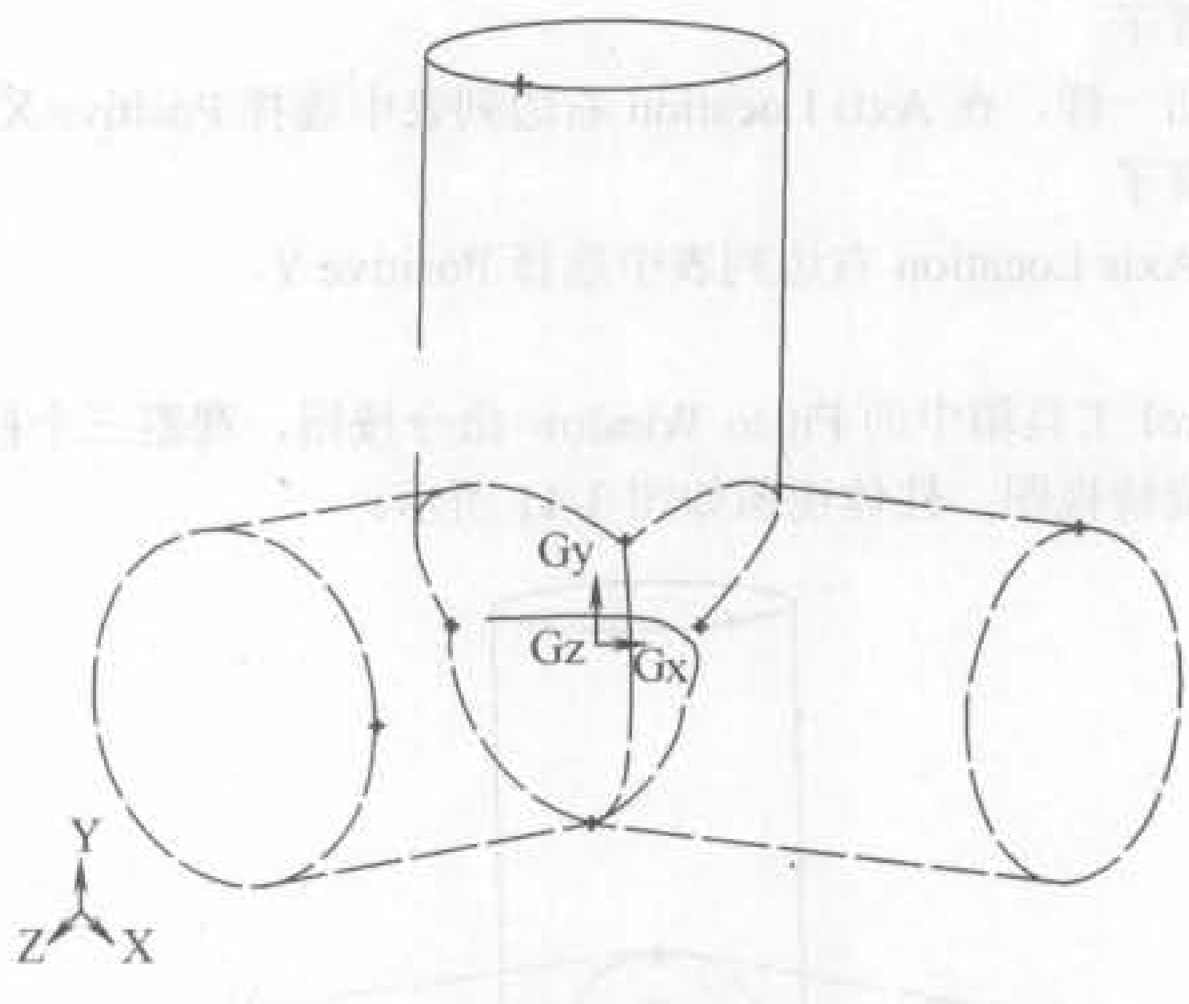


图 3-44 完整的几何体

3. 分解几何体

采用 TGrid 方法对几何体进行自动网格化是可能的。但是，用等角六面体对此几何体进行自动网格化则是不可能的。为了生成等角六面体网格，必须把这个几何体分解成几部分，然后对每一部分进行六面体网格划分。在这个例子中，将用一个方体来分割这个三管相贯体，在三管相贯处形成一个 1/8 球体，然后，再在几何体内创建一条边，用这条边创建三个面，最后用这三个面把三管相贯体分割成三根管子。

1) 创建一个方体

选择 Geometry→Volume→Create Volume (Brick)命令，打开如图 3-45 所示的 Create Real Brick 对话框。

(1) 在 Width 文本框中输入 5。

如果没有再输入值，Gambit 将把 Depth 和 Height 的值均设为 5。

(2) 在 Direction 右边的列表中选择-X-Y-Z。

(3) 单击 Apply 按钮。

生成的视图如图 3-46 所示。

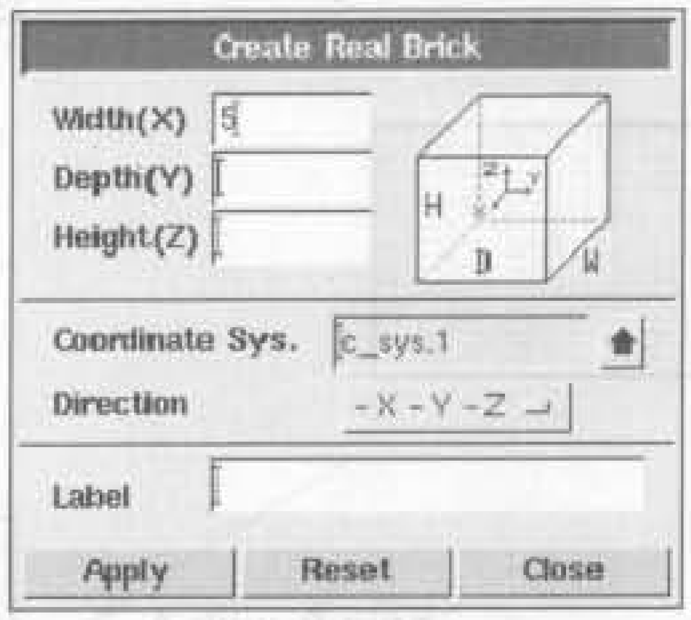


图 3-45　Create Real Brick 对话框

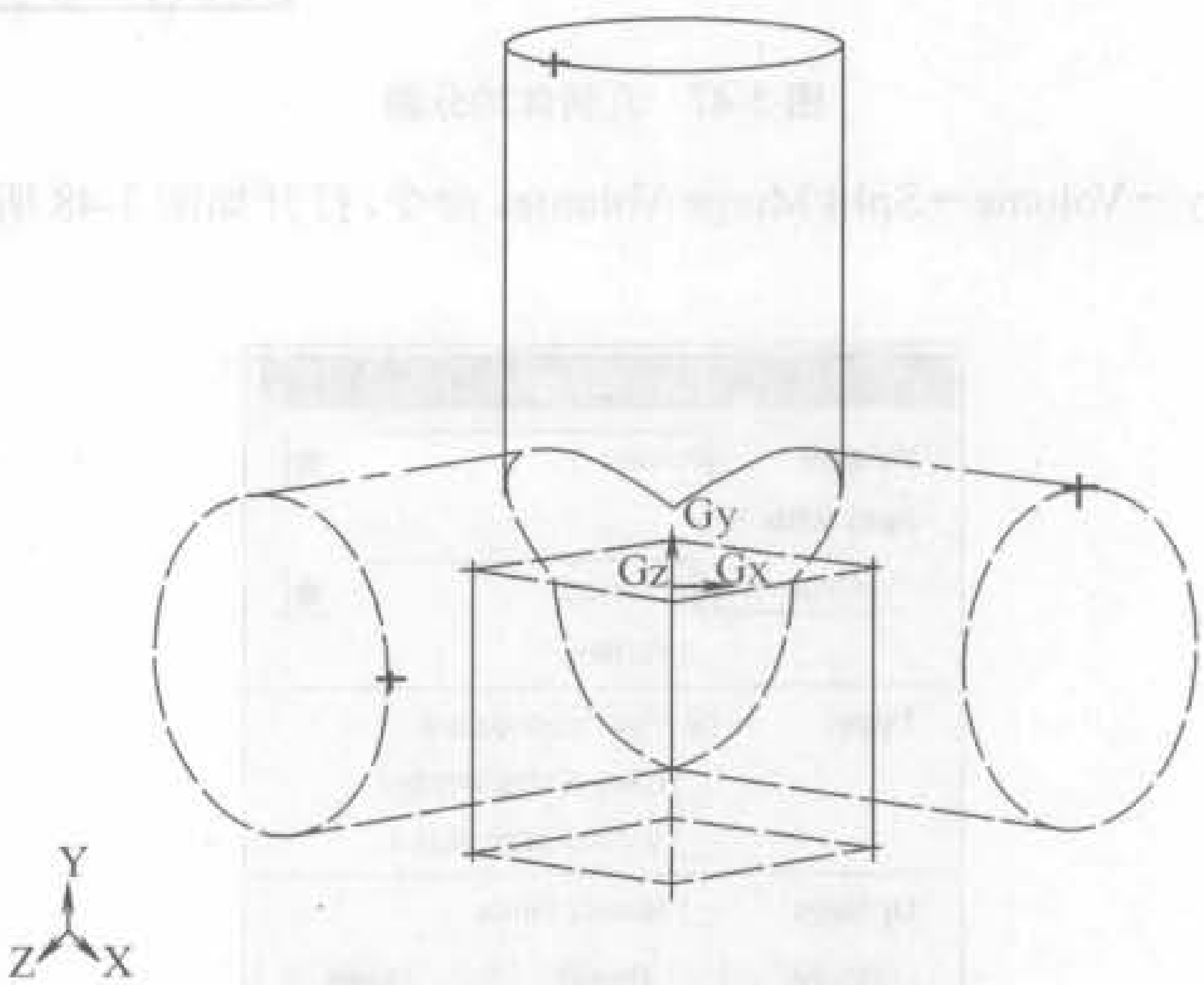


图 3-46　三管几何体和方体

2) 分割几何体，在三管相贯处形成一个 1/8 球体

如果用一个几何体来分割另一个几何体，将产生如下的几何体。

- 在相贯处产生与公共部位相对应的几何体。
- 产生与从第一个几何体减去第二个几何体的部位相对应的几何体。

换句话讲，分割一个几何体将产生一个相贯处与相减布尔操作的组合。选择几何体的顺序是很重要的。例如，图 3-47 中展示了用几何体 B 分割几何体 A 以及相反操作的差别。

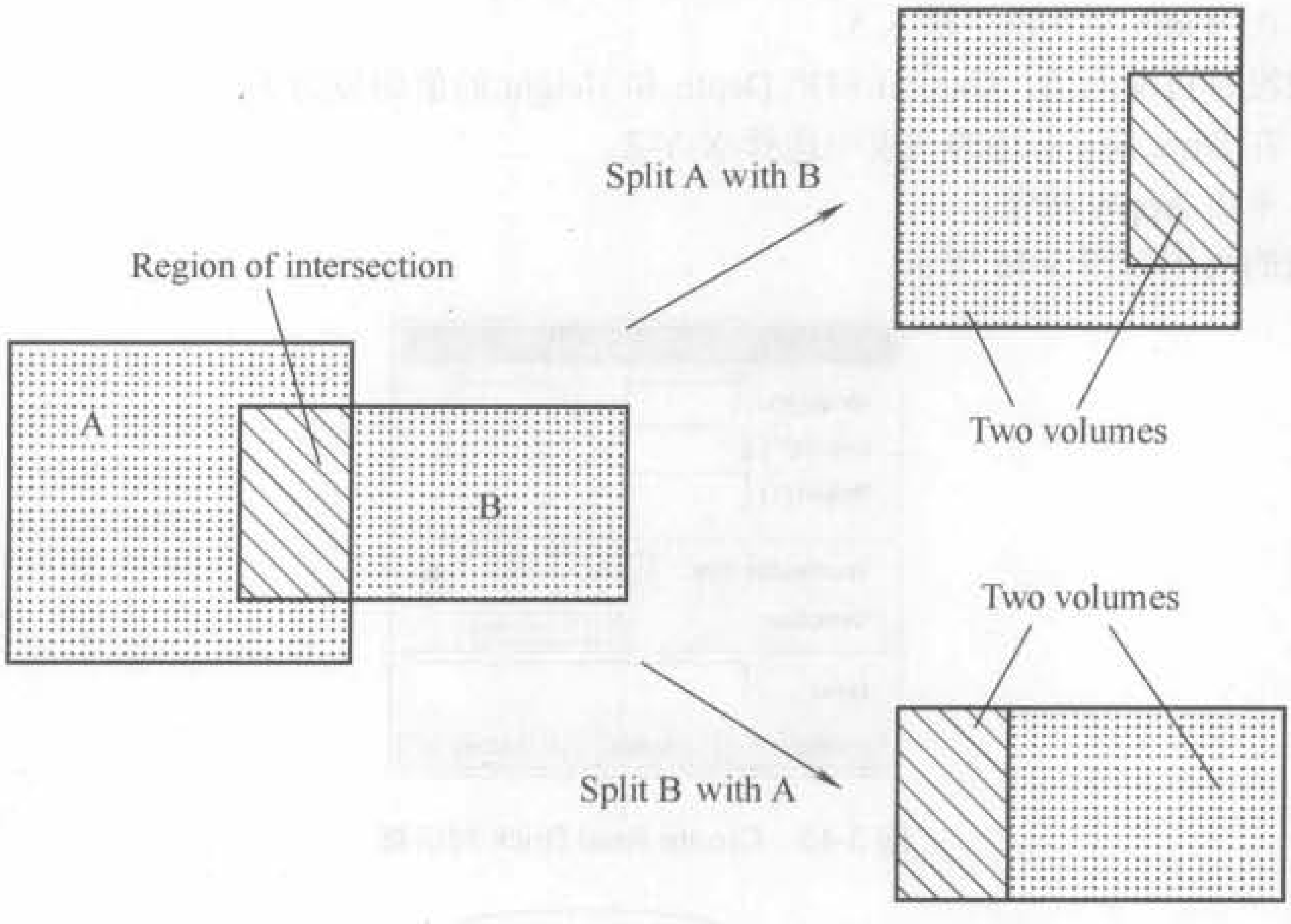


图 3-47 几何体的分割

选择 Geometry→Volume→Split/Merge Volumes 命令，打开如图 3-48 所示的 Split Volume 对话框。



图 3-48 Split Volume 对话框

(1) 在图形窗口中选择三管相贯体。

(2) 单击对话框中 Split With 右边的文本框，接受对三管相贯体的选择，并使 Split With 文本框激活。

(3) 选择方体，并单击 Apply 按钮。

Gambit 将用方体来分割三管相贯体，产生两个几何体：三管体(Volume.2)和 1/8 球体(Volume.3)。

3) 在三管相贯体中创建一条直边

选择 Geometry→Line→Create Edge 命令，打开如图 3-49 所示的 Create Straight Edge 对

话框。

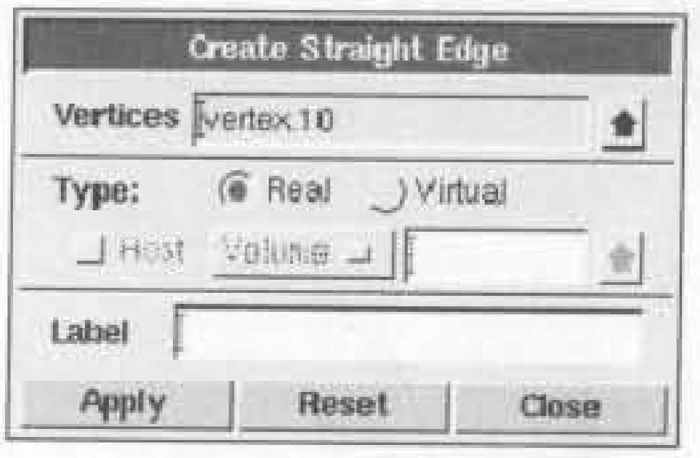


图 3-49　Create Straight Edge 对话框

(1)　单击原点(Gx,Gy,Gz)。

(2)　选择三个柱体共享的顶点(X=Y=Z)。

(3)　单击 Apply 按钮，接受顶点的选择，并在它们之间生成一条边。

该边在图 3-50 中呈黄色。

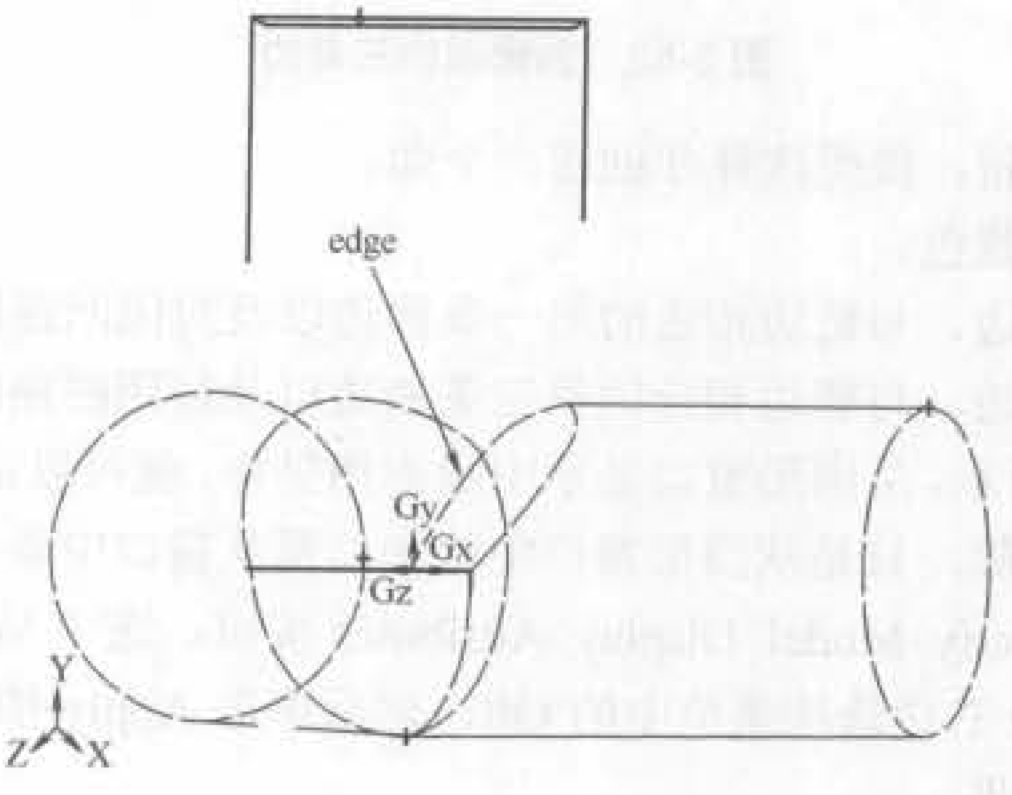


图 3-50　三管相贯体内创建的直边

4)　在三管相贯体中创建面

选择 Geometry→Face→Form Face 命令，打开如图 3-51 所示的 Create Face From Wireframe 对话框。

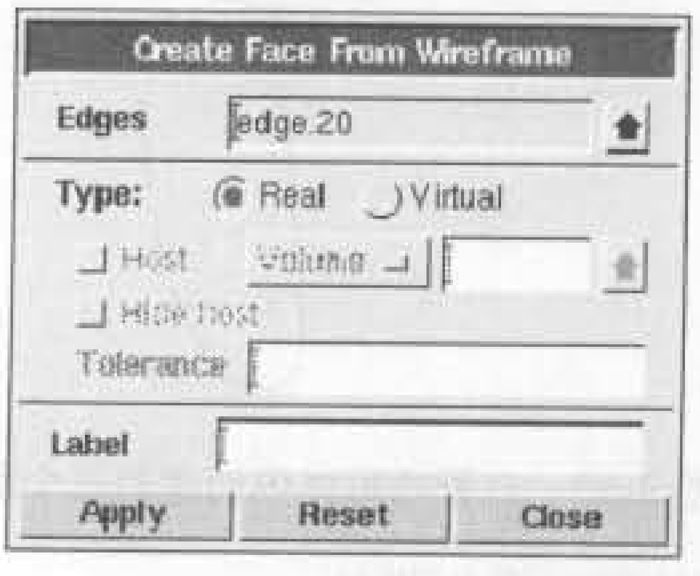


图 3-51　Create Face From Wireframe 对话框

(1)　用前面创建的边在相贯体内创建面

①　选择前面创建的边。

② 在其中一个柱体表面上选择与该边相连的一个曲边。

③ 选择一条边使回路封闭。

选择的三条边如图 3-52 所示。

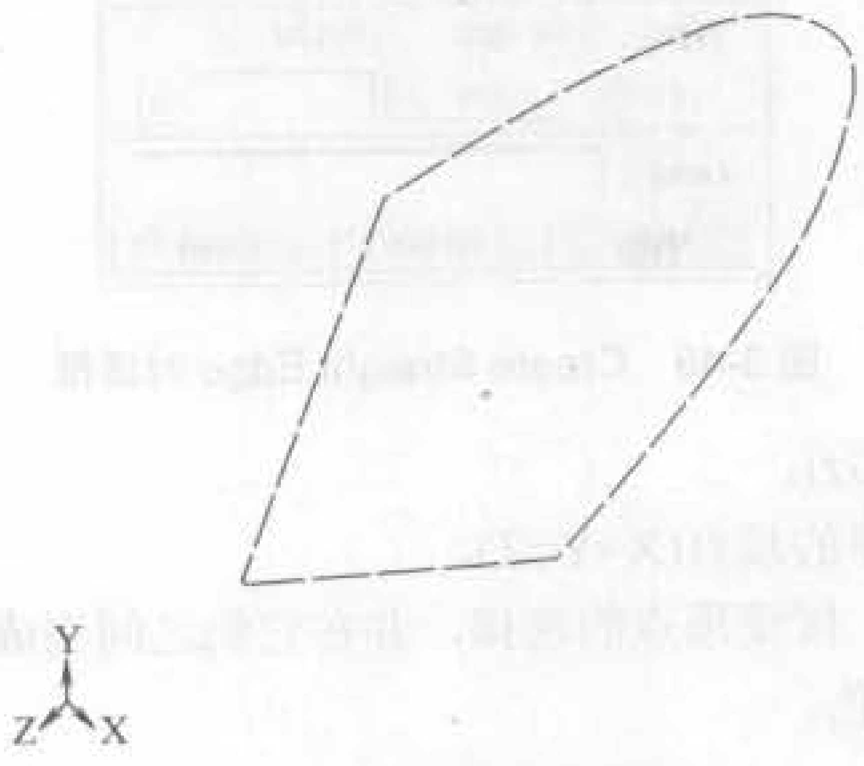


图 3-52 创建面的三条边

④ 单击 Apply 按钮，接受选择并创建一个面。

前面创建的边变成蓝色。

(2) 通过选择蓝色边、与蓝边相连的另一条曲边以及封闭回路的边来创建第二个面。

(3) 通过选择蓝色边、与蓝边相连的第三条曲边以及封闭回路的边来创建第三个面。

三个面如图 3-53 所示。从图形窗口显示中除去相贯体，就可以清楚看到创建的三个面。相贯体其实并没有被删除，只是从图形窗口中除去。要从窗口中除去相贯体，单击 Global Control 工具箱中的 Specify Model Display Attributes 按钮，选中 Volumes 复选框和 Visible 复选框，再选中 Visible 右边选项菜单中的 Off，然后单击 Apply 按钮。当考察完三个面之后，再使相贯体恢复可见。

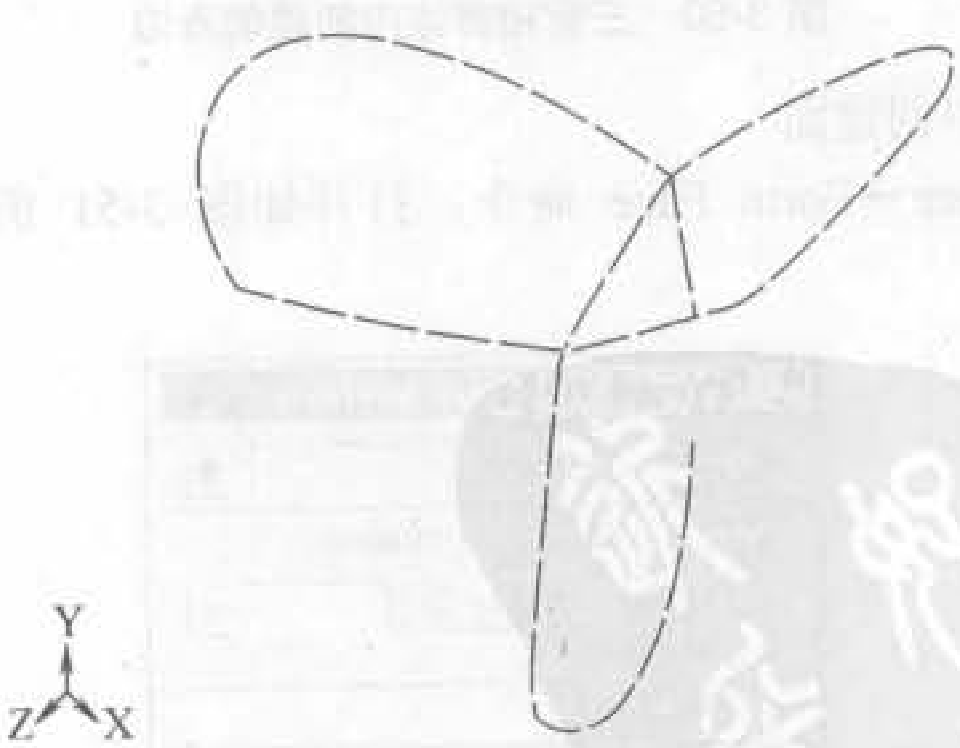


图 3-53 三管相贯体内创建的三个面

5) 用上面创建的三个面分割三管相贯体

选择 Geometry→Volume→Split/Merge Volumes 命令，打开如图 3-54 所示的 Split Volume 对话框。

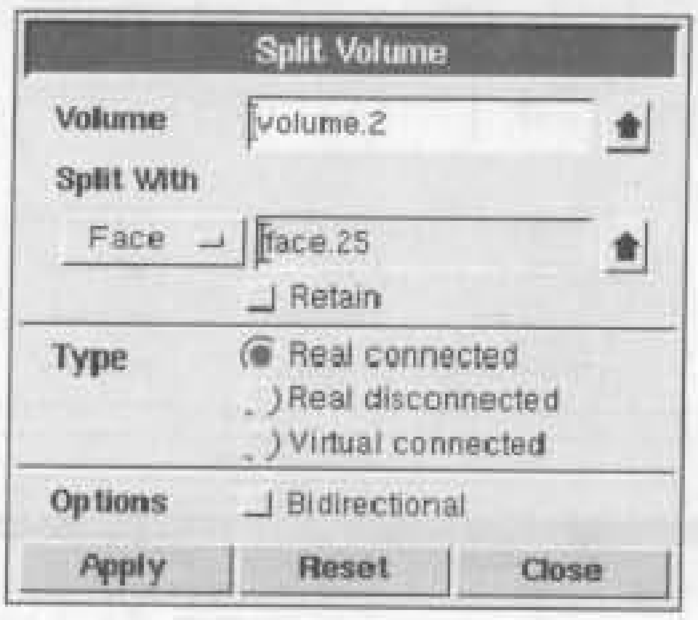


图 3-54 Split Volume 对话框

① 在图形窗口中选择三管相贯体。

② 在 Split With 下面的列表中选择 Face。

③ 选中前面创建的三个面中的一个。

如果想撤销选中的面并选择另一个面，用 Shift+鼠标中键选择面。

④ 单击 Apply 按钮，接受选择并分割相贯体。

Gambit 将用三个面(相互关联的)来把三管相贯体分割成三个小的几何体(分别代表三根管子)，这三个几何体都与公共几何体相连。分解后的几何体如图 3-55 所示。现在，就可以进行网格划分了。

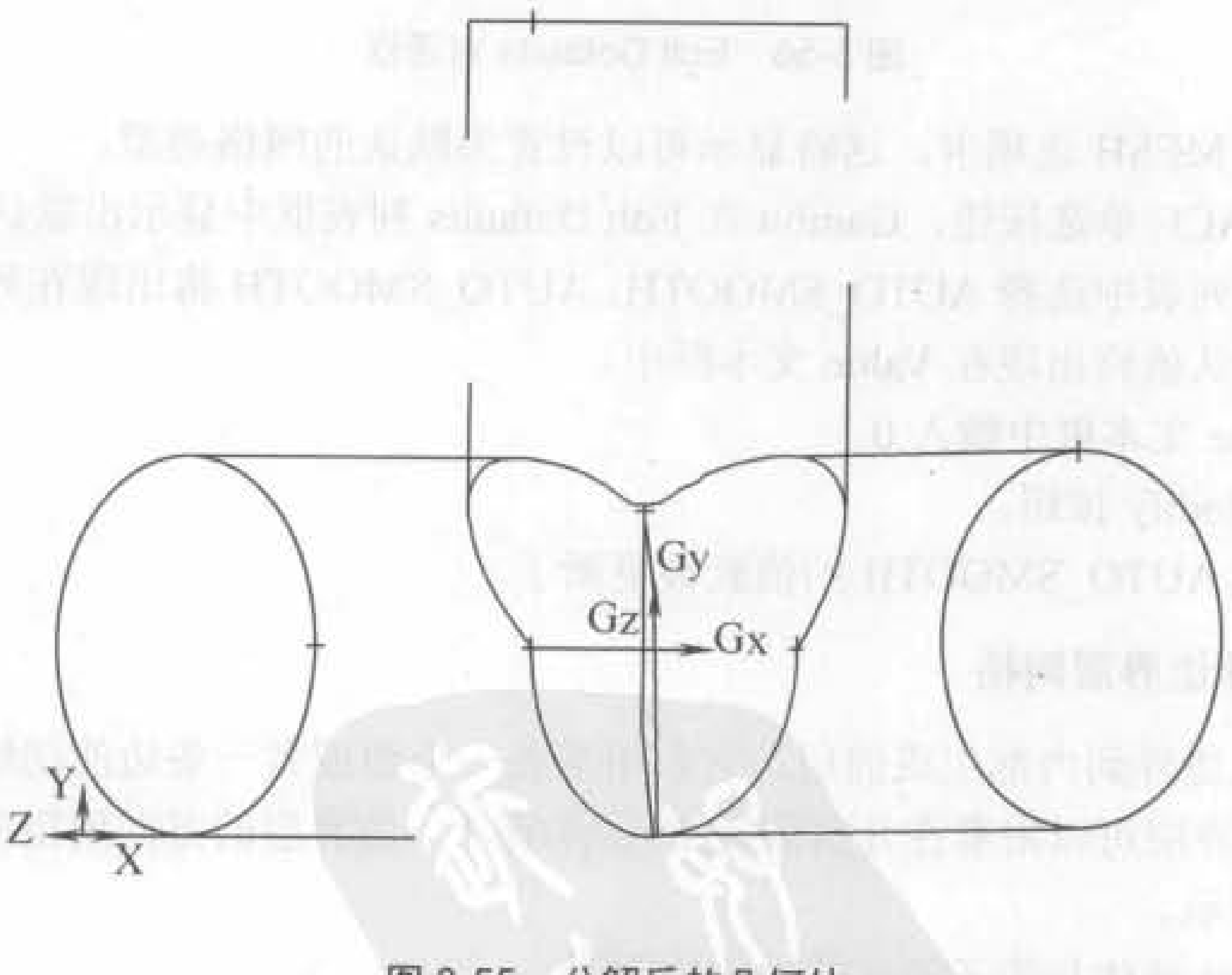


图 3-55 分解后的几何体

4. 关闭网格的自动光滑

在这个例子中，为了防止在几何体网格化时把边界层光滑掉，关闭网格的光滑是必要的。

选择 Edit→Defaults 命令，打开如图 3-56 所示的 Edit Defaults 对话框。

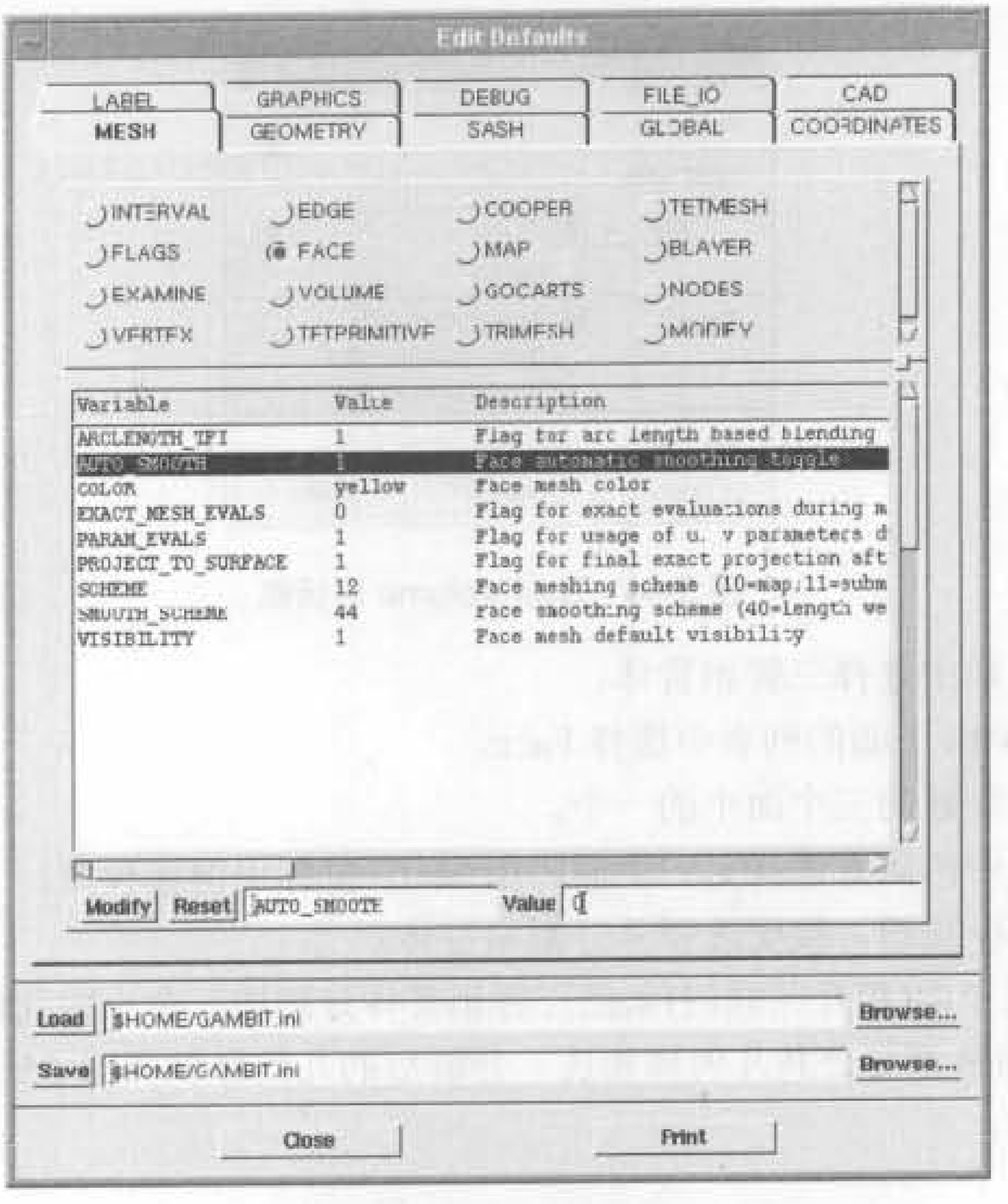


图 3-56 Edit Defaults 对话框

(1) 切换到 MESH 选项卡，这将显示可以设置为默认的网格类型。

(2) 选中 FACE 单选按钮，Gambit 在 Edit Defaults 列表框中显示出默认设置的变量。

(3) 在变量列表中选择 AUTO_SMOOTH，AUTO_SMOOTH 将出现在列表底部的一个文本框中，其默认值将出现在 Value 文本框中。

(4) 在 Value 文本框中输入 0。

(5) 单击 Modify 按钮。

列表中变量 AUTO_SMOOTH 的值就被更新了。

5. 壁面划分边界层网格

边界层是从边界到内部生成的层，它们用来在一个面或者一条边的法线方向上局部细化网格。单个边界层可以附着在几对面/边或者体/面上，边界层的方向用指向激活面或体的中心的箭头来表示。

1) 生成 1/8 球体与管子相交的边的边界层

选择 Mesh→Boundary Layer→Create Boundary Layer 命令，打开如图 3-16 所示的 Create Boundary Layer 对话框。

(1) 在 Definition 选项组中的 First row 文本框中输入 0.1，这定义了单元的第一行法线方向上的高度。

(2) 在 Growth factor 文本框中输入 1.4，这设置了单元续行之间的距离比。

(3) 移动 Rows 下面的滑块直到 Rows=4，这定义了单元行的总数。注意，Gambit 自动

更新 Depth，Depth 是指边界层的总高度。

(4) 保持 Transition pattern 的默认设置(1∶1)。

(5) 选择 1/8 球体与管子相交的三条曲边之一，如图 3-57 所示。边界层将在边上显示出来。

(6) 检查指示边界层方向的箭头是否指向原点(Gx,Gy,Gz)，如果没有，按下 Shift 键，用鼠标中键单击该边，直到箭头指向正确方向。

(7) 选择球体与管子相交的第二条曲边，并确保边上的箭头指向原点。

(8) 对第三条曲边重复上面的方法。边界层在边上的显示如图 3-57 所示。

(9) 单击 Apply 按钮，使得边界层应用于边上。

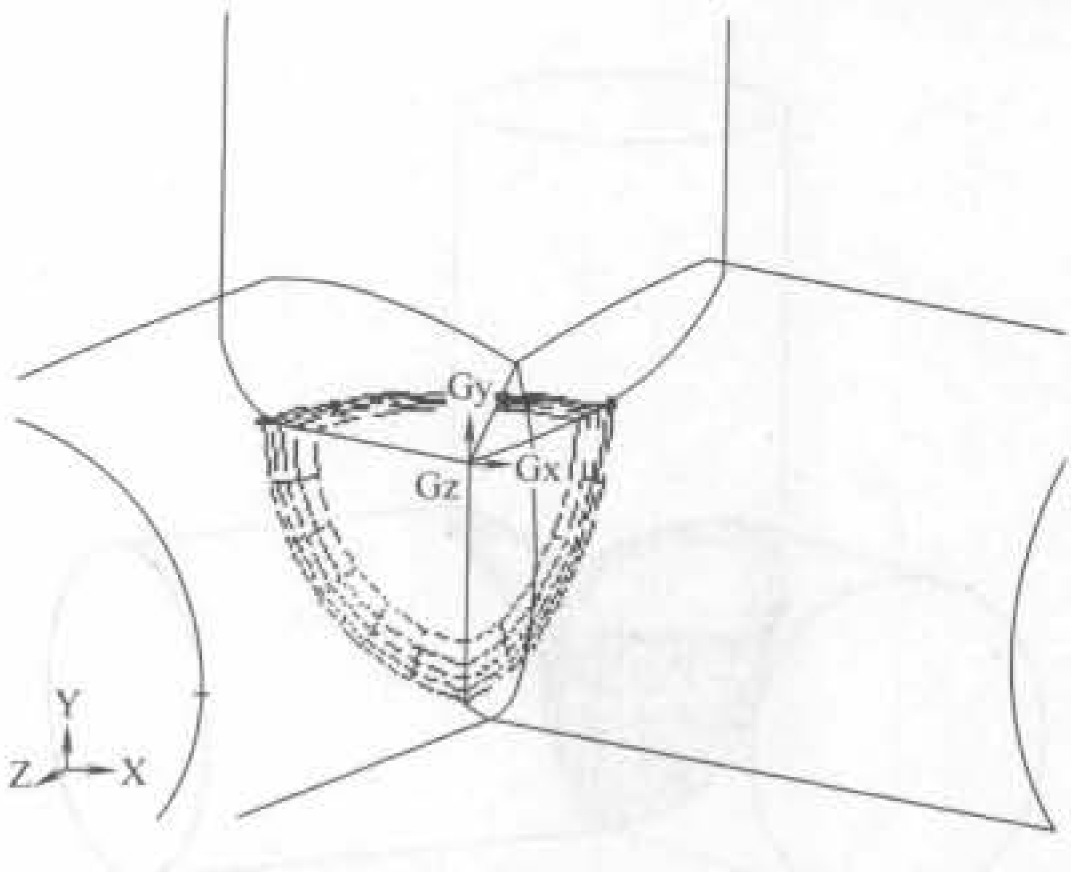


图 3-57　1/8 球体三边上的边界层

2) 其他部分边界层的生成

重复上面的步骤，在三管相交的曲边上生成边界层，如图 3-58 所示，同样，边上的箭头必须指向原点。

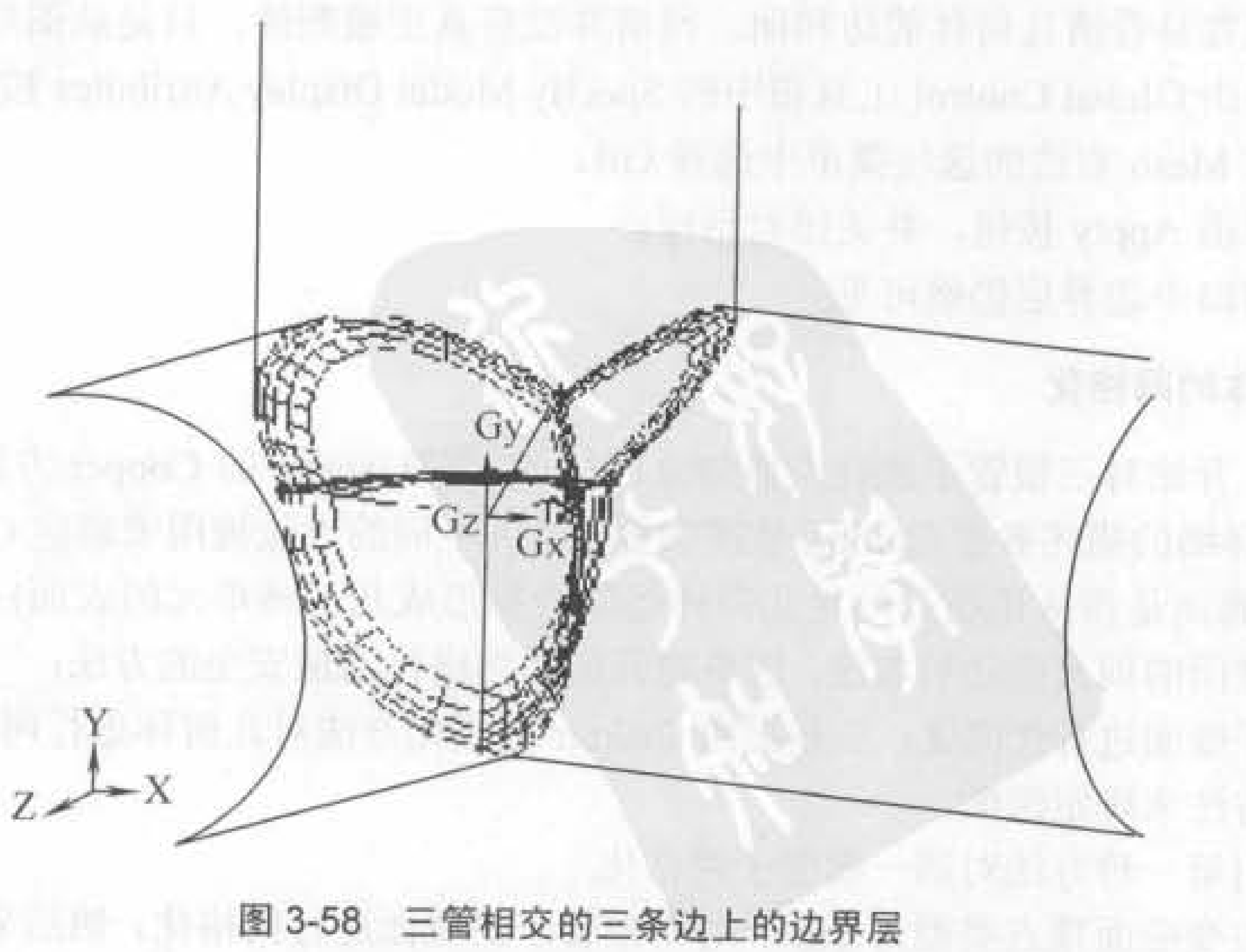


图 3-58　三管相交的三条边上的边界层

6. 对 1/8 球体进行网格划分

1) 球体网格化

选择 Mesh→Volume→Mesh Volumes 命令，打开如图 3-27 所示的 Mesh Volumes 对话框。

(1) 在图形窗口中选择球体。Gambit 自动选择 Scheme 下的 Hex Elements 和 Tet Primitive Type，这是因为这个几何体本身代表了一个逻辑四面体(注意：Tet Primitive 方法把一个逻辑四面体分为 4 个逻辑六面体块，并且在每一块内生成六面体网格单元，这种方法不会生成四面体网格)。

(2) Spacing 采用默认设置 Interval size，单击 Apply 按钮。1/8 球体的网格如图 3-59 所示，注意球体的三个面上的边界层。

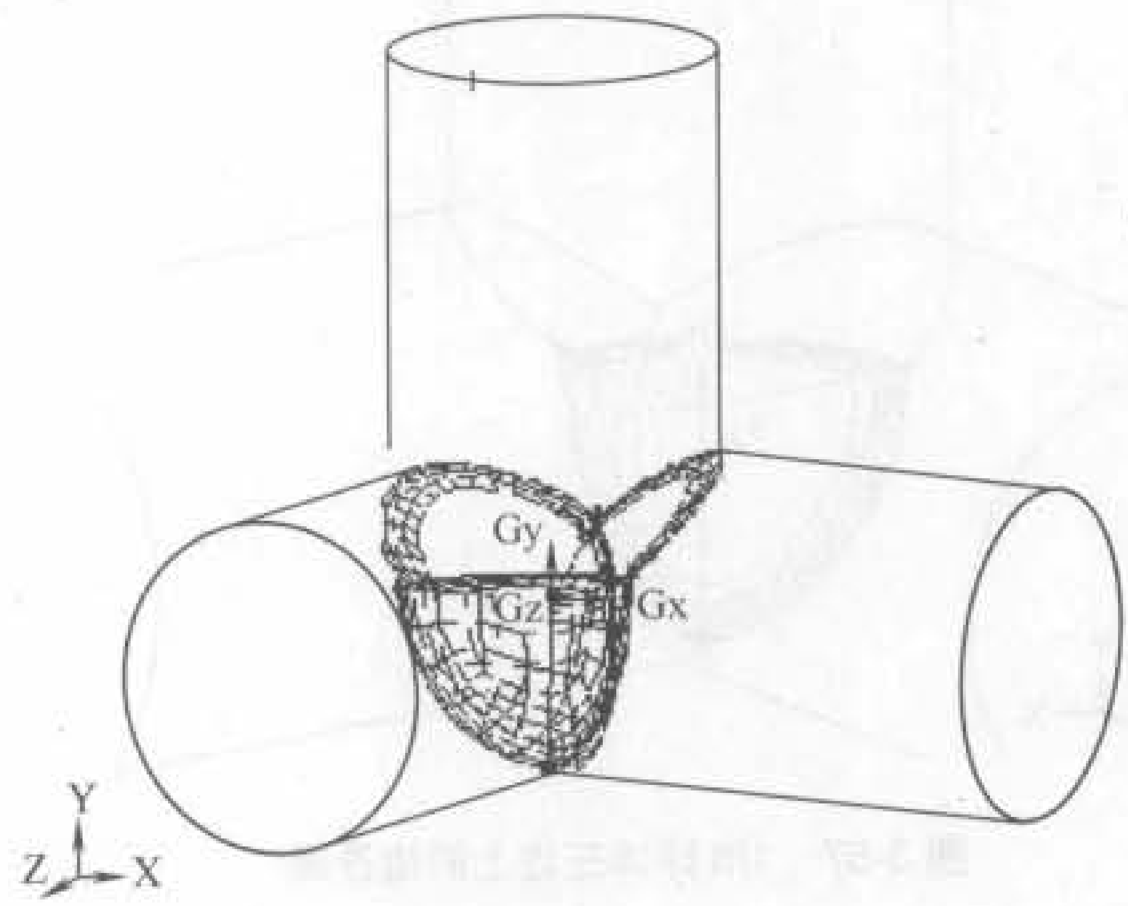


图 3-59　1/8 球体的网格

2) 在进行三管网格化之前除去已生成网格的显示

这样就容易看清几何体的边和面。网格并没有真正被删除，只是从图形窗口中除去。

(1) 单击 Global Control 工具箱中的 Specify Model Display Attributes 按钮。

(2) 在 Mesh 右边的选项菜单中选择 Off。

(3) 单击 Apply 按钮，并关闭对话框。

图形窗口中边界层仍然可见。

7. 柱体的网格化

现在，开始对三根管子进行网格划分，本例采用 Gambit 的 Cooper 方法对三根管子进行网格化(详细的描述参见 Gambit 建模向导)。三种不同的方法被用来确定 Cooper 方法所要求的源面(源面是指从该面开始对几何体进行扫描形成几何体单元的表面)：一是采用改变一根管子壁面的顶点类型的方法，这是确保正确网格化的最安全的方法；二是采用 Submap 方法对管子壁面进行网格化；三是采用 Cooper 网格化方法对几何体进行网格化，并采用手选源面的方法来确定源面。

1) 用第一种方法对第一根管子网格化

通过改变壁面顶点类型为 Side，对一根管子的壁面进行网格化，然后采用 Cooper 方法

对该柱体进行网格化

改变管子壁面的顶点类型为 Side 后，就可以使 Gambit 在壁面上运用 Submap 方法，这样，Cooper 网格化方法的规范就可以在管子上实现，进而实现对该管体的网格化。

(1) 改变壁面顶点类型为 Side

选择 Mesh→Face→Set Face Vertex Type 命令，打开如图 3-60 所示的 Set Face Vertex Type 对话框。

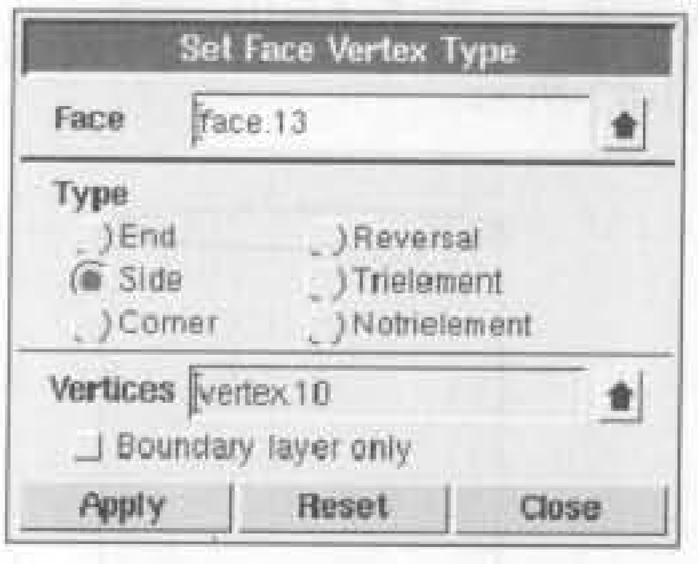


图 3-60　Set Face Vertex Type 对话框

① 在对话框中的 Type 选项组中选中 Side(默认值)单选按钮。

② 在图形窗口中选择一根管子的壁面，如图 3-61 所示。注意，壁面上标为 E 的顶点代表三根管子的交点。

③ 单击 Vertices 文本框。

④ 在图形窗口中选择标有 E 的顶点(三管交点，如图 3-61 所示)。

⑤ 单击 Apply 按钮。

该顶点最后变为 S，代表 Side，如果重新选择面，将只会看到顶点标签。同时一条消息出现在 Transcript 窗口中，说明顶点被设置成了 Side。

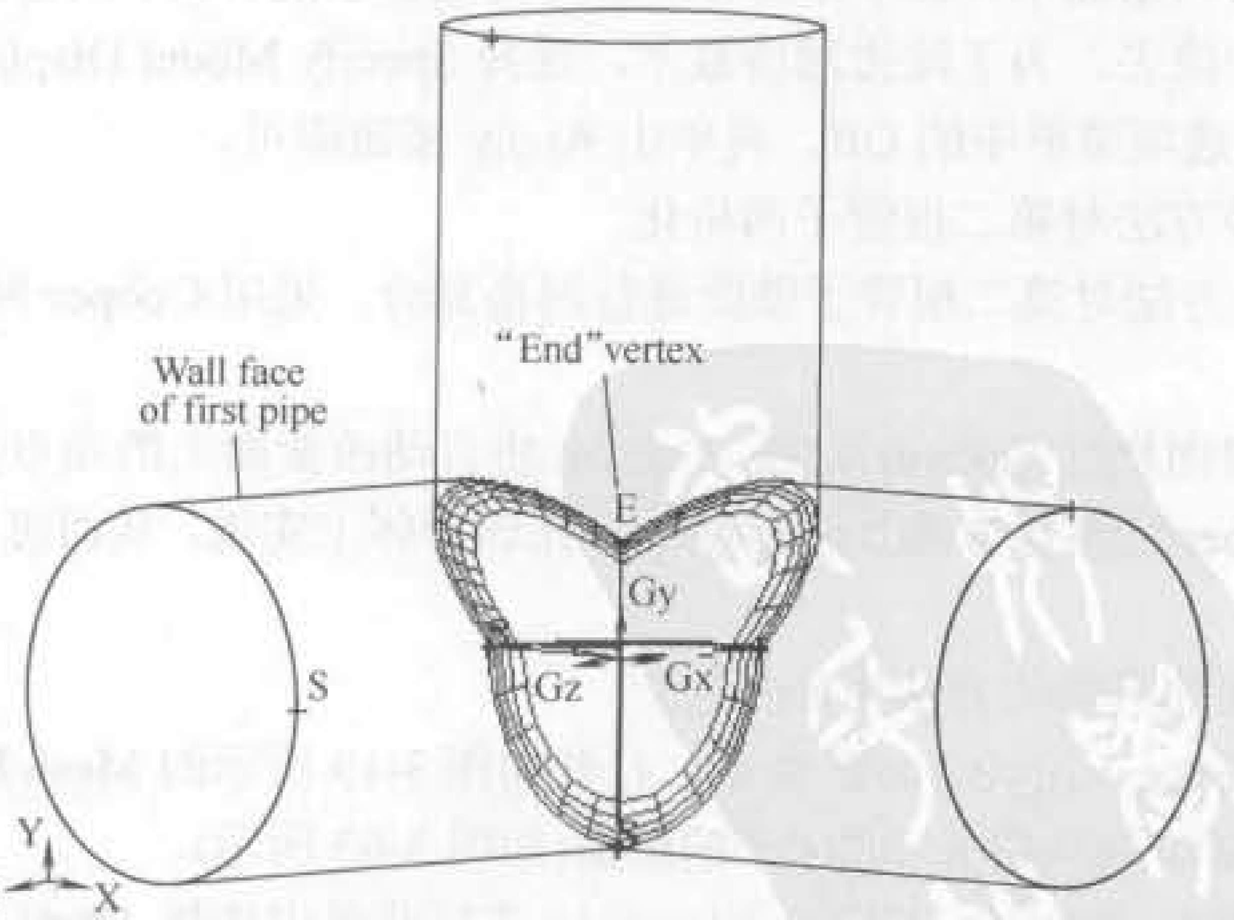


图 3-61　第一根管子壁面上的 End 顶点

(2) 用 Cooper 网格化方法对管体进行网格化

选择 Mesh→Volume→Mesh Volumes 命令，打开如图 3-27 所示的 Mesh Volumes 对话框。

① 在图形窗口中选择第一个管体。

注意，在对话框中 Scheme 下自动选择了 Hex Elements 和 Cooper Type，这是因为壁面顶点已经变为 Side。

因为壁面顶点类型改变为 Side，Gambit 将自动选择源面。

② 保留 Interval size 默认设置 1，单击 Apply 按钮。

该管子便被网格化，如图 3-62 所示。

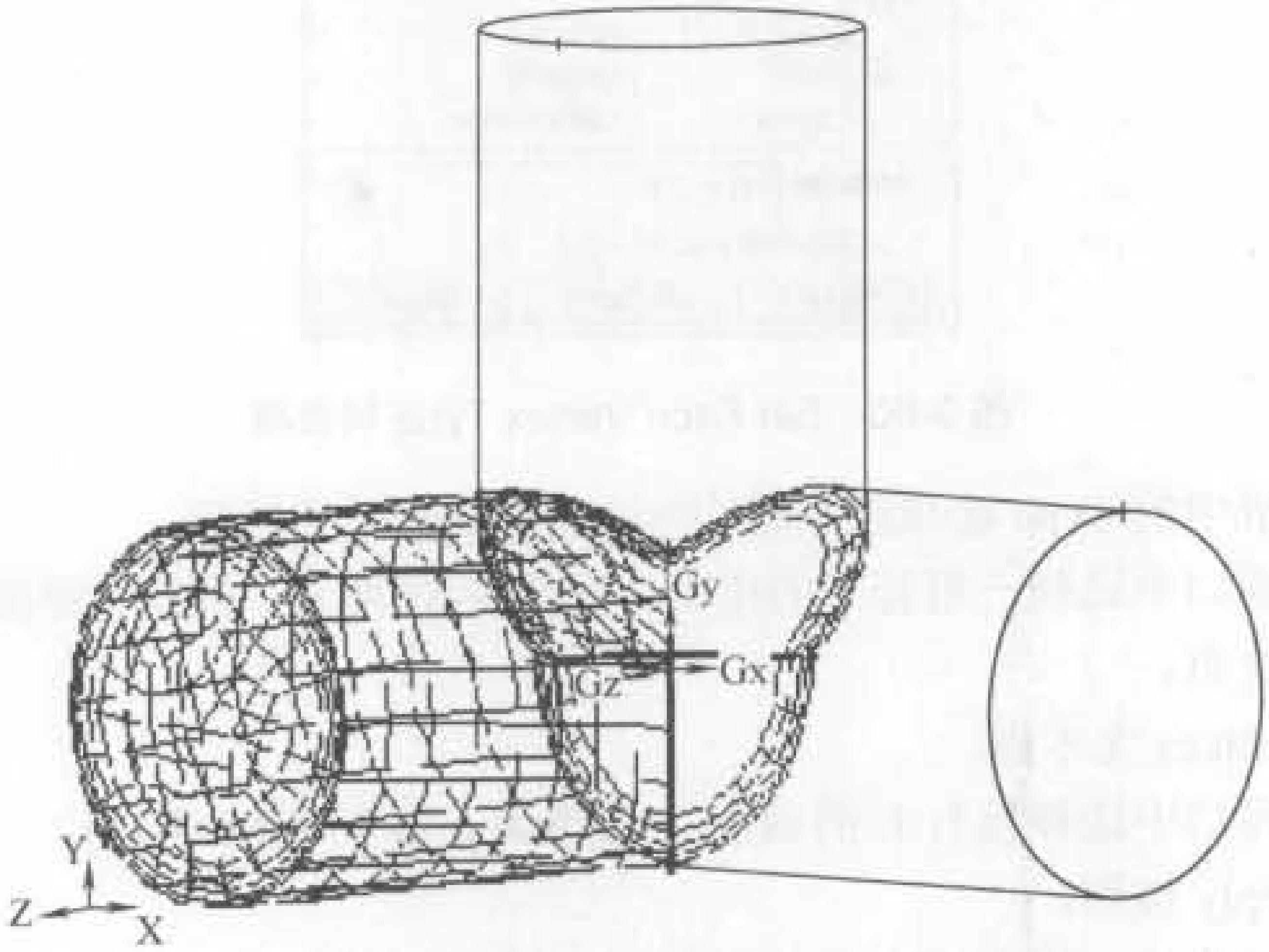


图 3-62　Cooper 网格化结果

这时最好除去网格显示，这样容易看清另外两根管子的各面。网格并不是真的被删除，只是从图形窗口中除去。为了除去网格显示，选择 Specify Model Display Attributes 命令，选择 Mesh 右边的选项菜单中的 Off，再单击 Apply 按钮即可。

2) 用第二种方法对第二根管子网格化

采用 Submap 方法对第二根管子壁面进行网格划分，采用 Cooper 网格化方法对管体进行网格化

通过对管子壁面执行 Submap 方法，Gambit 将自动改变面上的顶点类型去迎合 Submap 方法，然后，Cooper 网格化方法的规范就可以在该管体上实现，从而可以用 Cooper 方法对其进行网格划分。

(1) 设置壁面的网格化方法为 Submap

选择 Mesh→Face→Mesh Faces 命令，打开如图 3-19 所示的 Mesh Faces 对话框。

① 在图形窗口中选择第二根管子的壁面(如图 3-63 所示)。

② 在对话框中 Scheme 下面的 Elements 选项菜单中选择 Quad，在 Type 下面选择 Submap。

关于 Submap 方法的更多信息，参阅 Gambit 建模向导。

③ 保留 Interval size 的默认设置 1。

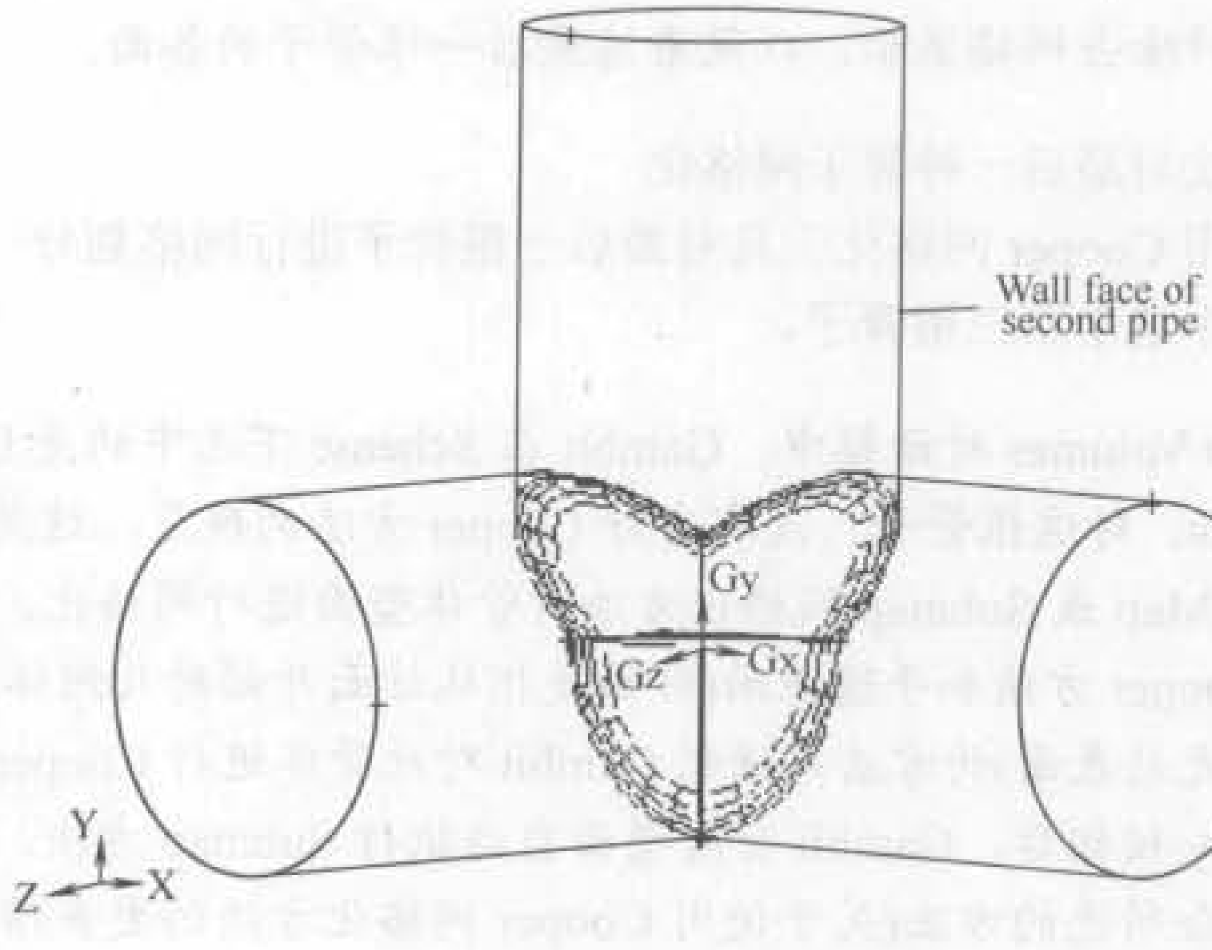


图 3-63　第二个管体的壁面

④　取消选中 Options 下面的 Mesh 复选框。

不选中 Mesh 复选框是因为这时不想对面进行网格划分，只是想在该面上应用这种网格化方法。Gambit 将在网格化管体时采用具体的方法对该面进行网格化。

⑤　单击 Apply 按钮。

(2) 用 Cooper 网格化方法对管体进行网格划分

选择 Mesh→Volume→Mesh Volumes 命令，打开如图 3-27 所示的 Mesh Volumes 对话框。

①　在图形窗口中选中管体。

注意，在对话框中的 Scheme 下自动选择了 Hex Elements 和 Cooper Type，这是因为对该管壁面采用了 Submap 方法。

由于对管壁面采用了 Submap 方法，Gambit 将自动选择源面。

②　保留 Interval size 的默认值 1，单击 Apply 按钮。

网格化后的管子如图 3-64 所示。

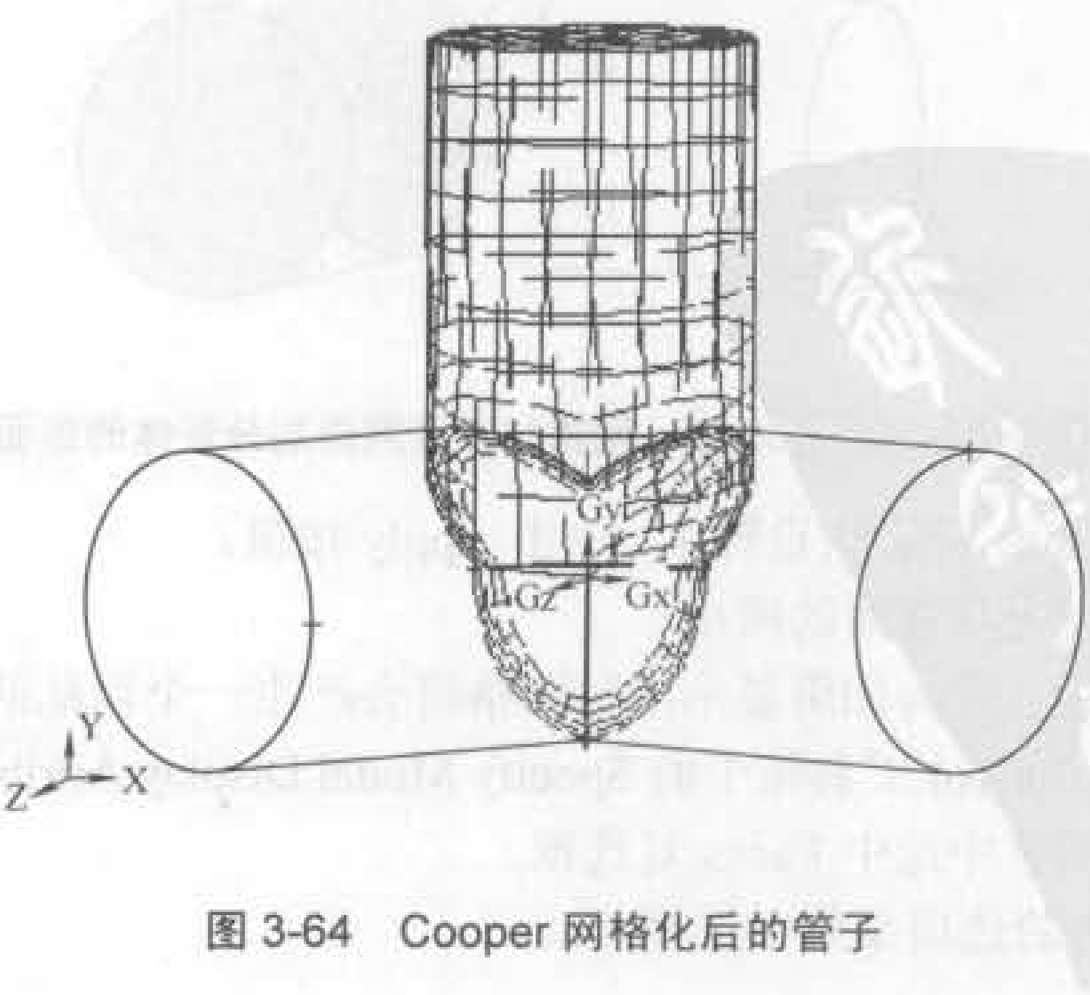


图 3-64　Cooper 网格化后的管子

注意： 此时最好除去网格显示，以便看清最后一根管子的各面。

3) 用第三种方法对最后一种管子网格化

通过手选源面，用 Cooper 网格化工具对最后一根管子进行网格划分

(1) 在图形窗口中选中第三根管子。

注意： 在 Mesh Volumes 对话框中，Gambit 在 Scheme 下选中的是 Undetermined Elements。对这根管子，没有实行 Cooper 方法的规范，这是因为 Gambit 不能采用 Map 或 Submap 网格化方法对管体壁面进行网格化。然而，可以通过选择 Cooper 方法和手选源面(源面是指从该面开始对几何体进行扫描形成几何体单元的表面)的方法来强制 Gambit 对此管体进行 Cooper 网格划分。当单击 Apply 按钮后，Gambit 会对壁面自动执行 Submap 方法，以及改变顶点类型去迎合所选的方法(关于使用 Cooper 网格化方法的更多信息，参见 Gambit 建模向导)。

(2) 在对话框中的 Scheme 下的 Elements 选项菜单中选择 Hex，Type 下选择 Cooper。

(3) 单击 Sources 列表框，然后，在图形窗口中选择除该管管壁外的所有面作为网格划分的源面，这些面如图 3-65 所示的 A、B、C、D。

按 Shift+鼠标中键单击一个面来撤销对该面的选择，再选择另外的面，也可以单击 Reset 按钮来撤销对所有面和体的选择，重新设置所有的参数。

选中的 4 个面在管子的两端，如图 3-65 所示。这可以在图形窗口中进行选择，也可以在 Sources 的列表中进行选择。

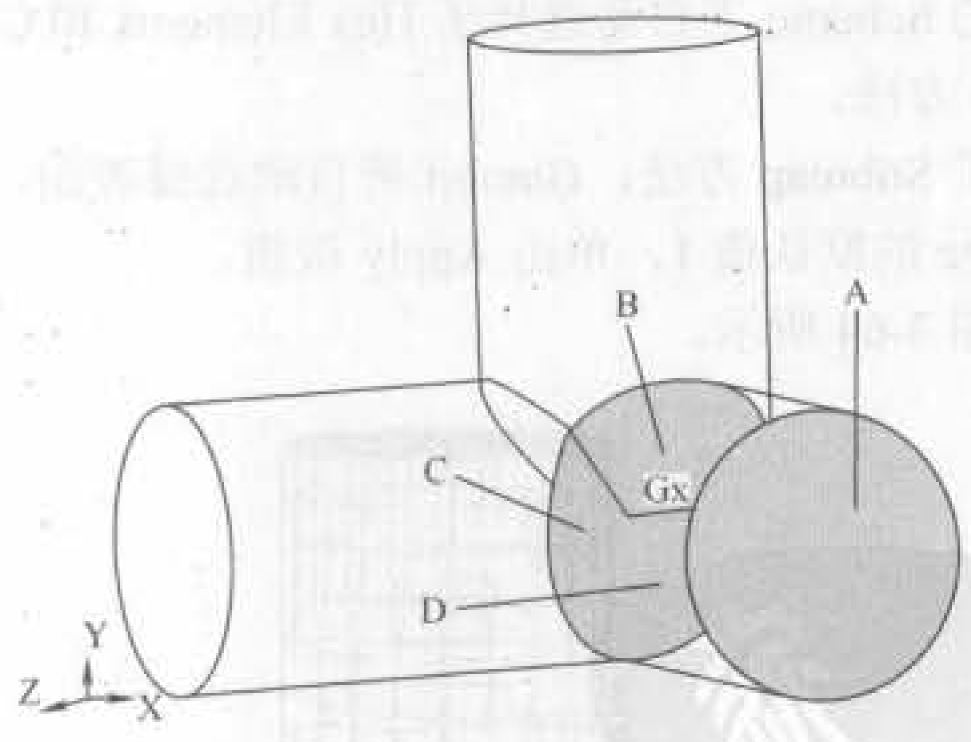


图 3-65 运用 Cooper 网格化方法网格划分管体的源面

(4) 保留 Interval size 的默认设置 1，单击 Apply 按钮。

4) 显示三管相贯体所有面的网格

这里只显示面网格，因为如果显示出体网格将会产生一个混乱的视图。

(1) 单击 Global Control 工具箱中的 Specify Model Display Attributes 按钮。

(2) 在打开的对话框中选中 Faces 复选框。

(3) 在 Mesh 右边的选项菜单中选择 On。

(4) 单击 Apply 按钮，并关闭该对话框。

三管相贯体的面网格如图 3-66 所示。

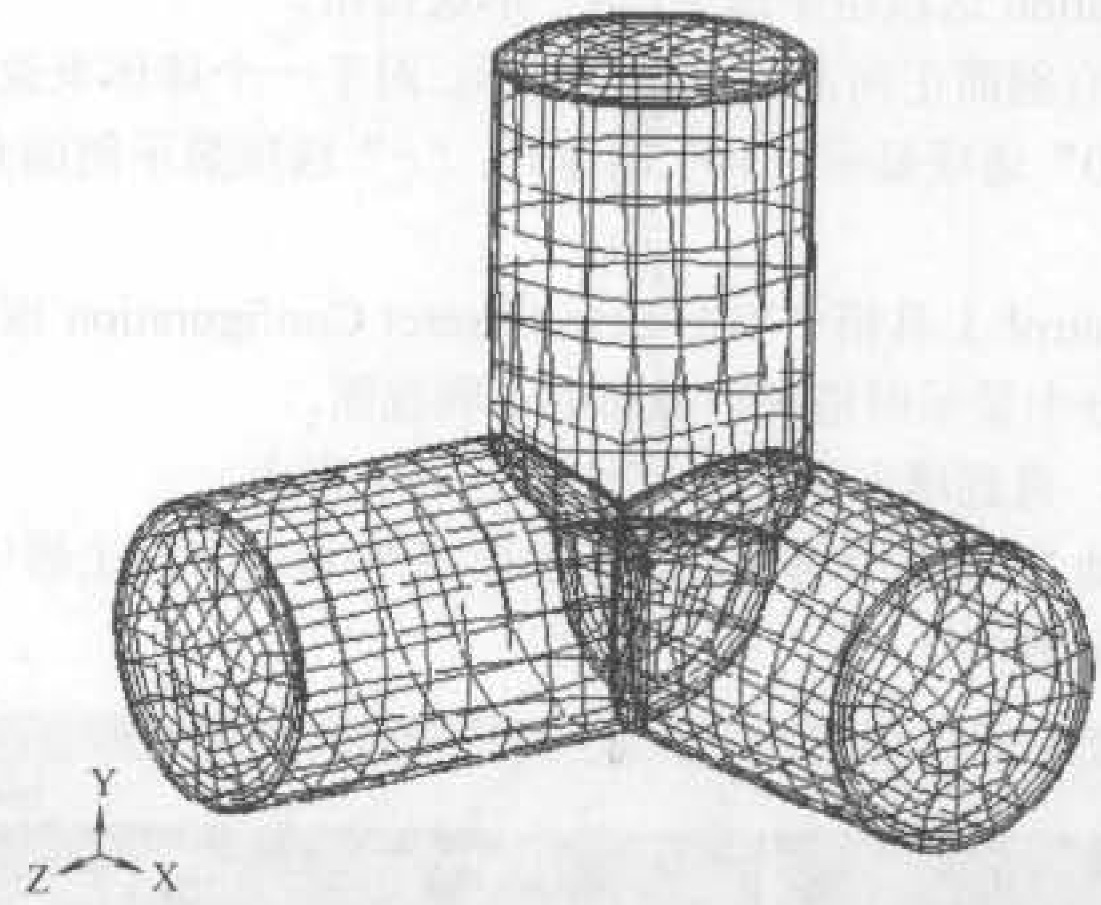


图 3-66　三管相贯体的面网格

8. 检查网格的品质

在 Global Control 工具箱中单击 Examine Mesh 命令按钮，打开如图 3-67 所示的 Examine Mesh 对话框。

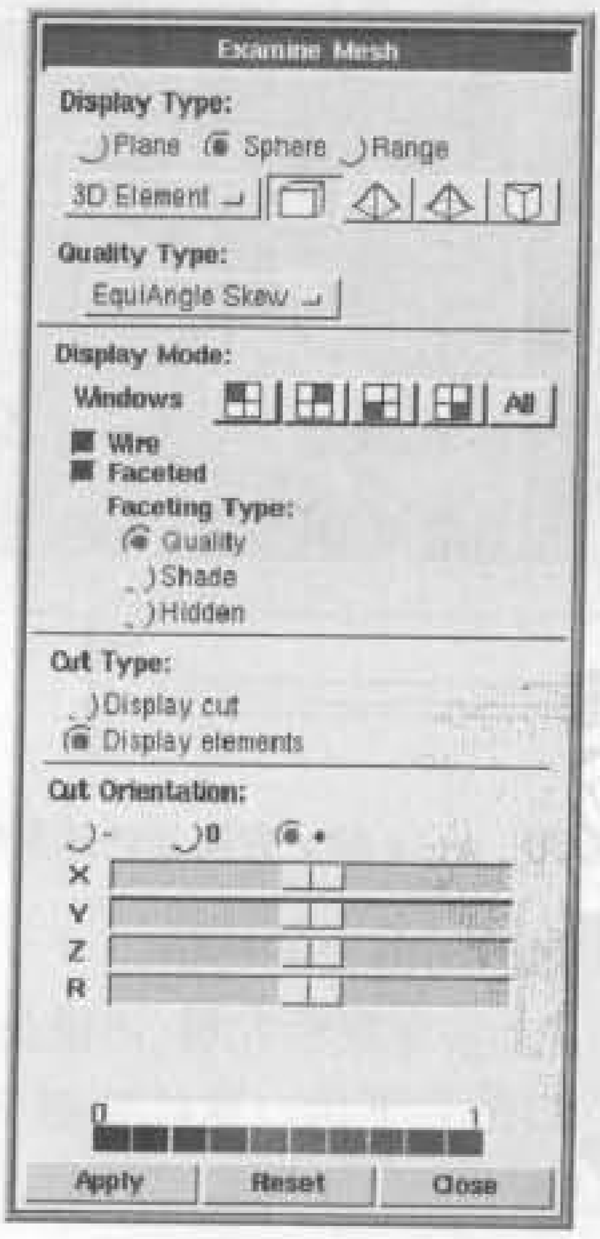


图 3-67　Examine Mesh 对话框

(1) 在对话框中的 Display Type 选项组中选中 Sphere 单选按钮。这就生成一个球形网格截面。对于三管相贯体，球形截面要比平面截面显示出更多的有用信息。

对话框中的 3D Element 类型是默认设置，选中其后面的方体。

(2) 从 Quality Type 选项菜单中选择 EquiAngle Skew。

(3) 在 Cut Orientation 选项组中选中“+”单选按钮。

“+”选项表明只有剖面正向上的单元被显示。对于一个球体来说，这就意味着只有球体内部的单元可见；“0”选项显示剖面上的单元；“-”选项显示剖面负向上的单元(即球体外部的单元)。

(4) 在 Global Control 工具箱中单击 Select Preset Configuration 按钮。这把图形窗口分成 4 部分，在每一部分中显示网格球形截面的不同视图。

(5) 拖动 X 滑块，直到球形剖面在 X 方向上居中为止。

(6) 同样地，移动 Y 和 Z 滑块使得球形剖面在 Y 和 Z 方向上居中。

视图显示如图 3-68 所示。

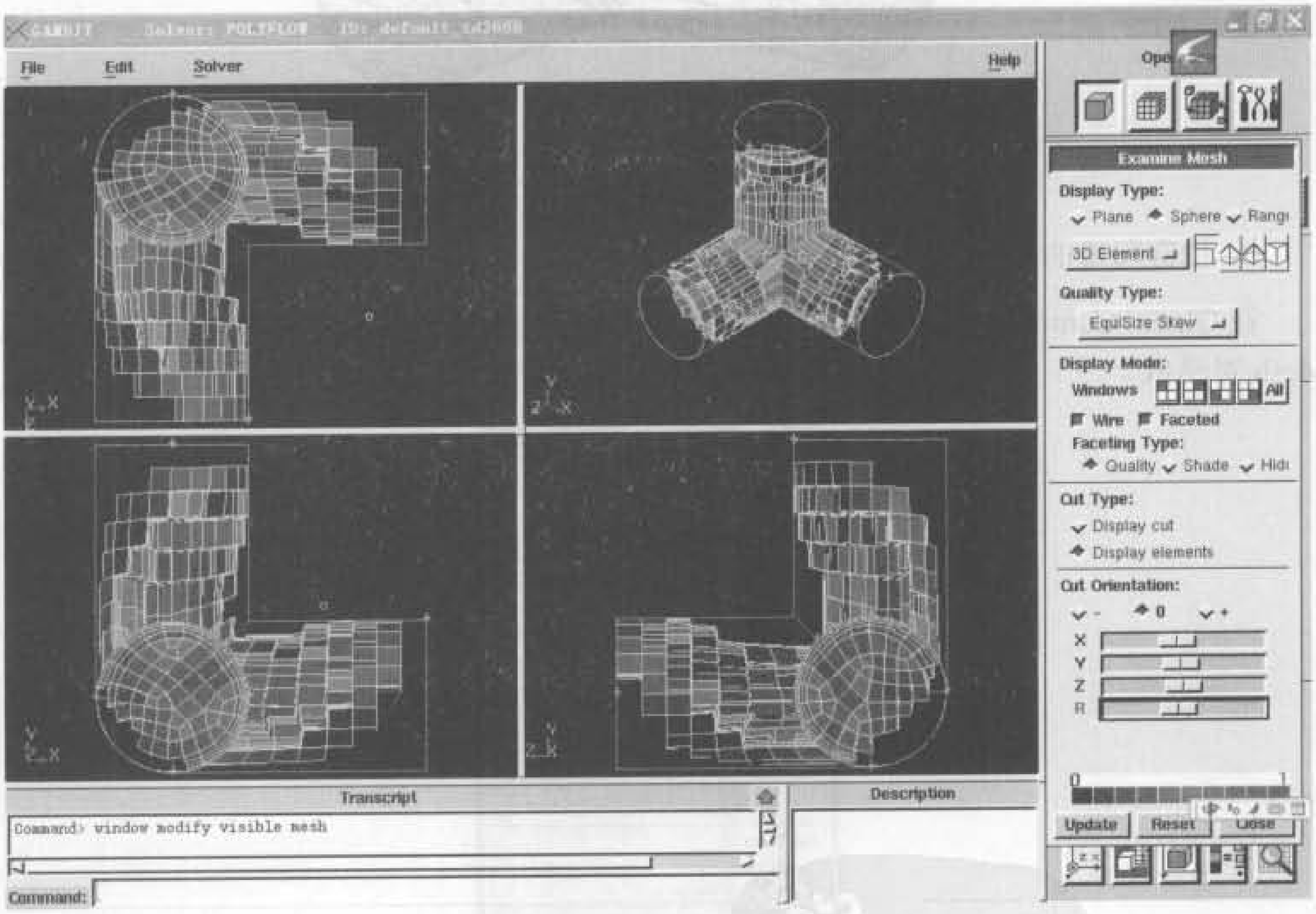


图 3-68　x、y 和 z 方向上的球形剖面

(7) 移动 R 滑块来观察不同球形剖面上的网格。

(8) 在图形窗口中的 Sash 锚上按住鼠标左键(该锚是指位于图形窗口中心的一个灰色小块)，对角地拖动该锚到图形窗口的右下角，这样就剩下一个单窗口。

(9) 关闭 Examine Mesh 对话框。

网格的球形截面即被除去，面网格也被储存起来。

9. 设置边界类型

1)　设置边界类型之前，除去网格和边界层显示

这样容易看清几何体的各边和各面。网格和边界层并没有被删除，只是从图形窗口中除去。

(1) 在 Global Control 工具箱中单击 Specify Display Attributes 按钮，弹出如图 3-69 所示的 Specify Display Attributes 对话框。

(2) 在 Mesh 右边的选项菜单中选择 Off。

(3) 单击 Apply 按钮。

(4) 选中 B.Layers 复选框，在 Visible 右边的选项菜单中选择 Off。

(5) 单击 Apply 按钮，并关闭对话框。

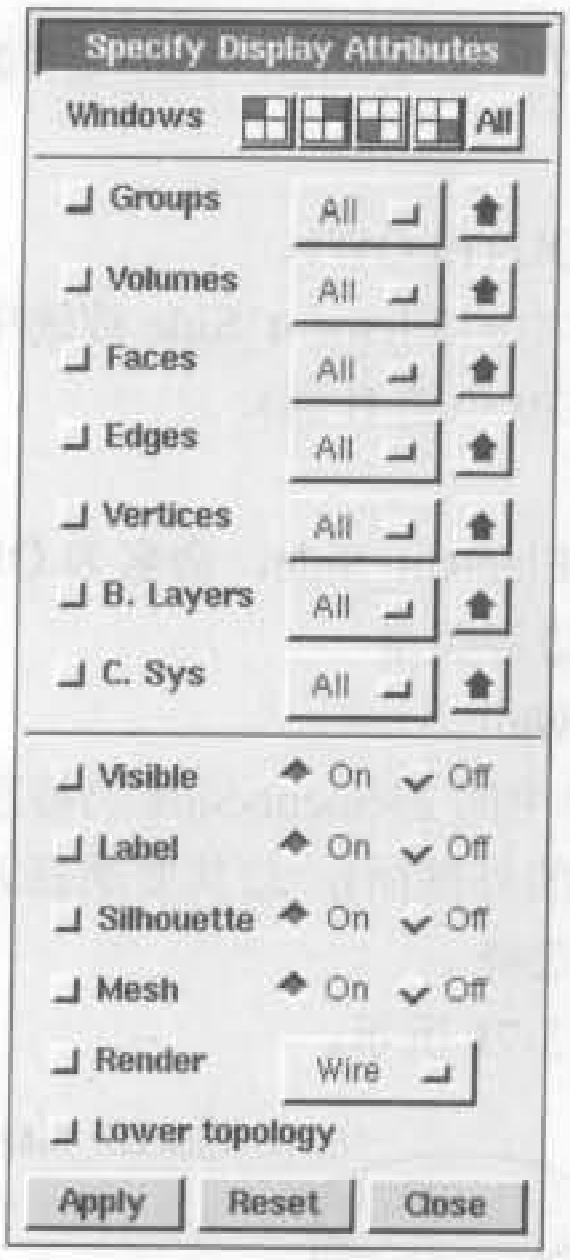


图 3-69　Specify Display Attributes 对话框

2)　设置三管相贯体的边界类型

选择 Zone→Specify Boundary Types 命令，打开如图 3-21 所示的 Specify Boundary Types 对话框。

(1) 定义入口流

① 在 Name 文本框中输入 INFLOW。

② 在 Type 选项菜单中选择 Element-Side。具体的边界类型将在 POLYFLOW 解算器中定义。

③ 检查确定 Entity 下的 Faces 被选中。

④ 在其中一根管子的尾端单击(如图 3-70 所示的 A 面)，然后单击 Apply 按钮。

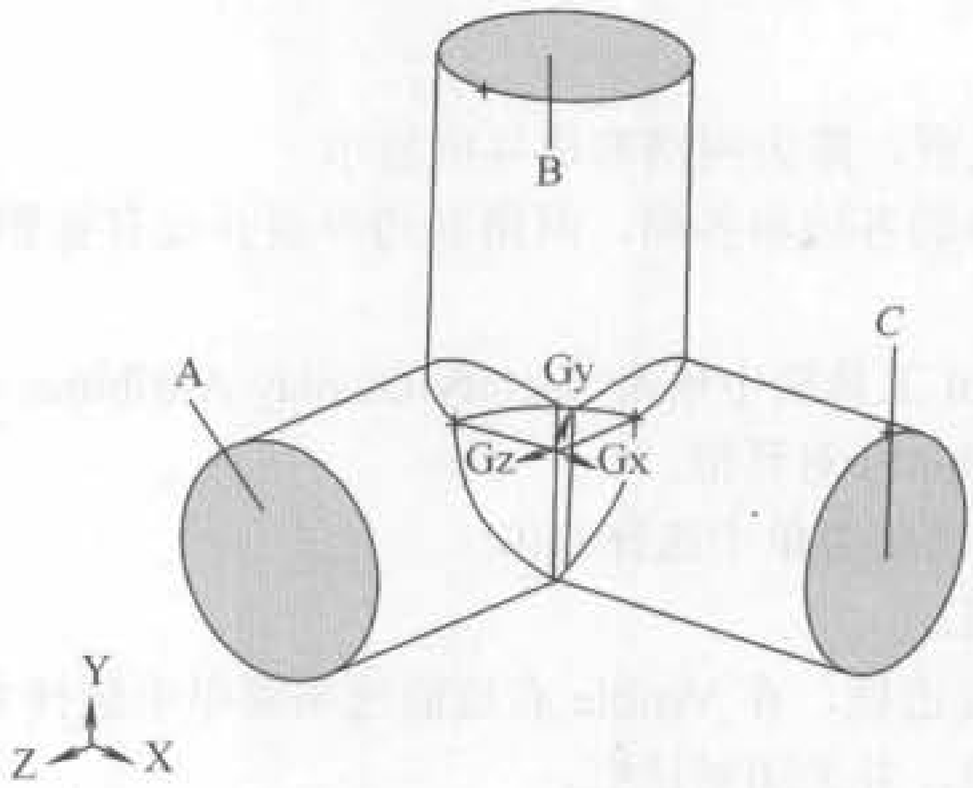


图 3-70 三管相贯体的面的边界类型

(2) 定义两个出口流

① 在 Name 文本框中输入 OUTFLOW1。

② 检查确定 Type 选项菜单中的 Element_Side 被选中，然后在图形窗口中选中剩下两管的其中一根管子的尾端(如图 3-70 中的 B 面)。

③ 单击 Apply 按钮。

④ 设置第三根管子尾端为 Element_Side，命名为 OUTFLOW 2。

(3) 定义三管相贯体的壁面边界类型

① 在 Name 文本框中输入 Wall。

② 检查确定 Type 选项菜单中的 Element-Side 仍被选中，然后在图形窗口中选中所有的壁面(三管的外壁面和 1/8 球体的外壁面)，总共要选择四个面。

③ 单击 Apply 按钮，接受选择。

三管相贯体的边界设置如图 3-71 所示。

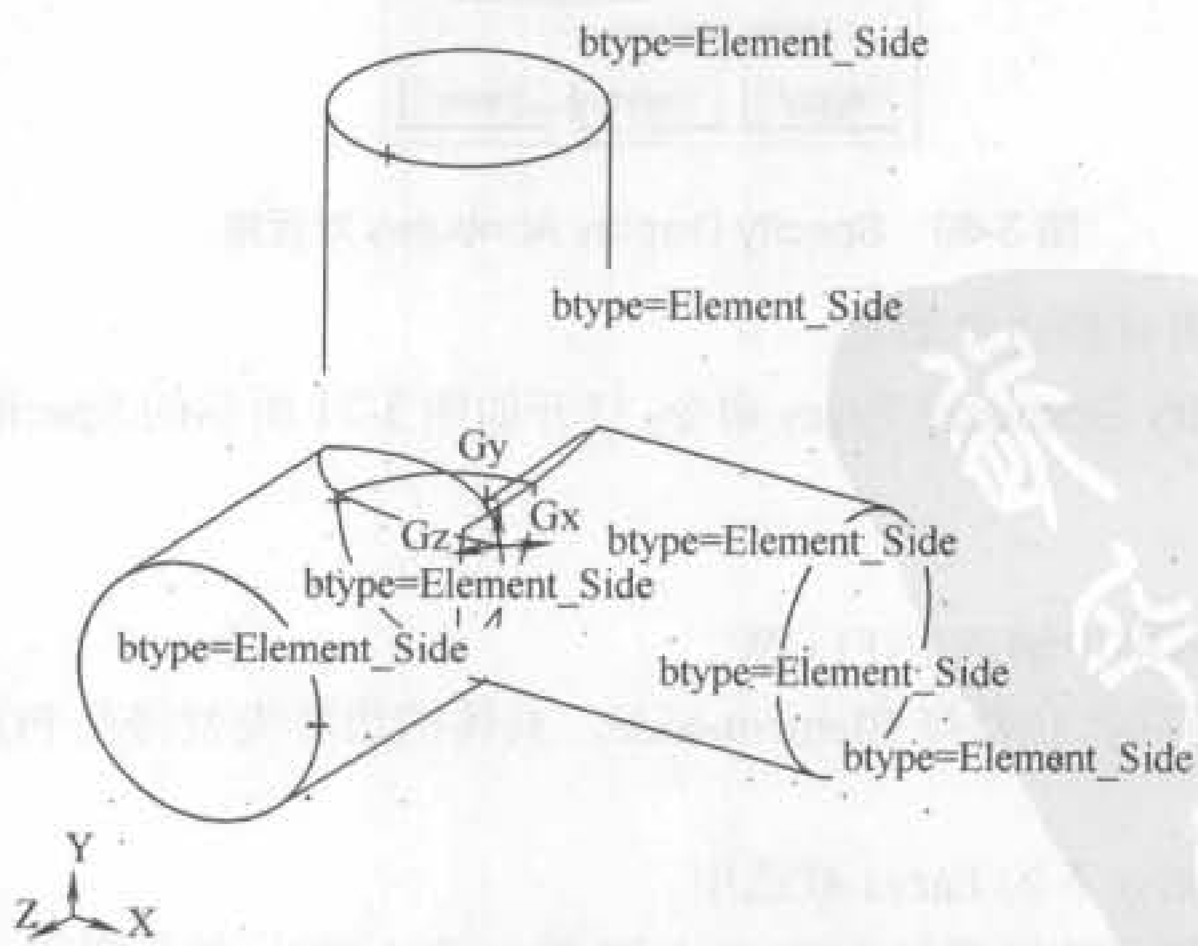


图 3-71 三管相贯体的边界类型

注意： 当 Gambit 写一个网格时，对任何几何体(3D)，如果没有确定连续介质类型，则将被默认设为 FUILD，这意味着在这里不需在 Specify Continuum Types 面板中确定连续介质类型。

10. 导出网格，保存任务

(1) 导出三管相贯体的网格文件

选择 File→Export→Mesh 命令，打开 Export Mesh File 对话框，如图 3-37 所示。

① 在对话框中输入导出文件的文件名。

② 单击 Accept 按钮。文件将被写入到工作目录下。

(2) 保存 Gambit 任务，并退出 Gambit

选择 File→Exit 命令，Gambit 将提示是否希望在退出之前保存当前任务，单击 Yes 按钮，保存当前任务并退出 Gambit。

第 4 章　通用后处理 Tecplot 使用入门

Tecplot 是 Amtec 公司推出的一个功能强大的科学绘图软件，它提供了丰富的绘图格式，包括 X-Y 曲线图，多种格式的二维和三维面绘图，以及三维体绘图格式。该软件易学易用，界面友好。而且针对 Fluent 软件有专门的数据接口，可以直接读入.cas 和.dat 文件，也可以在 Fluent 软件中选择输出的面和变量，然后直接输出 Tecplot 格式文档。

4.1 Tecplot 基本功能

4.1.1　Tecplot 的界面

如图 4-1 所示，为在加载二维数据的情况下 Tecplot 10.0 的开始界面，界面可以分成 4 个区：菜单栏、工具栏、状态栏和工作区。

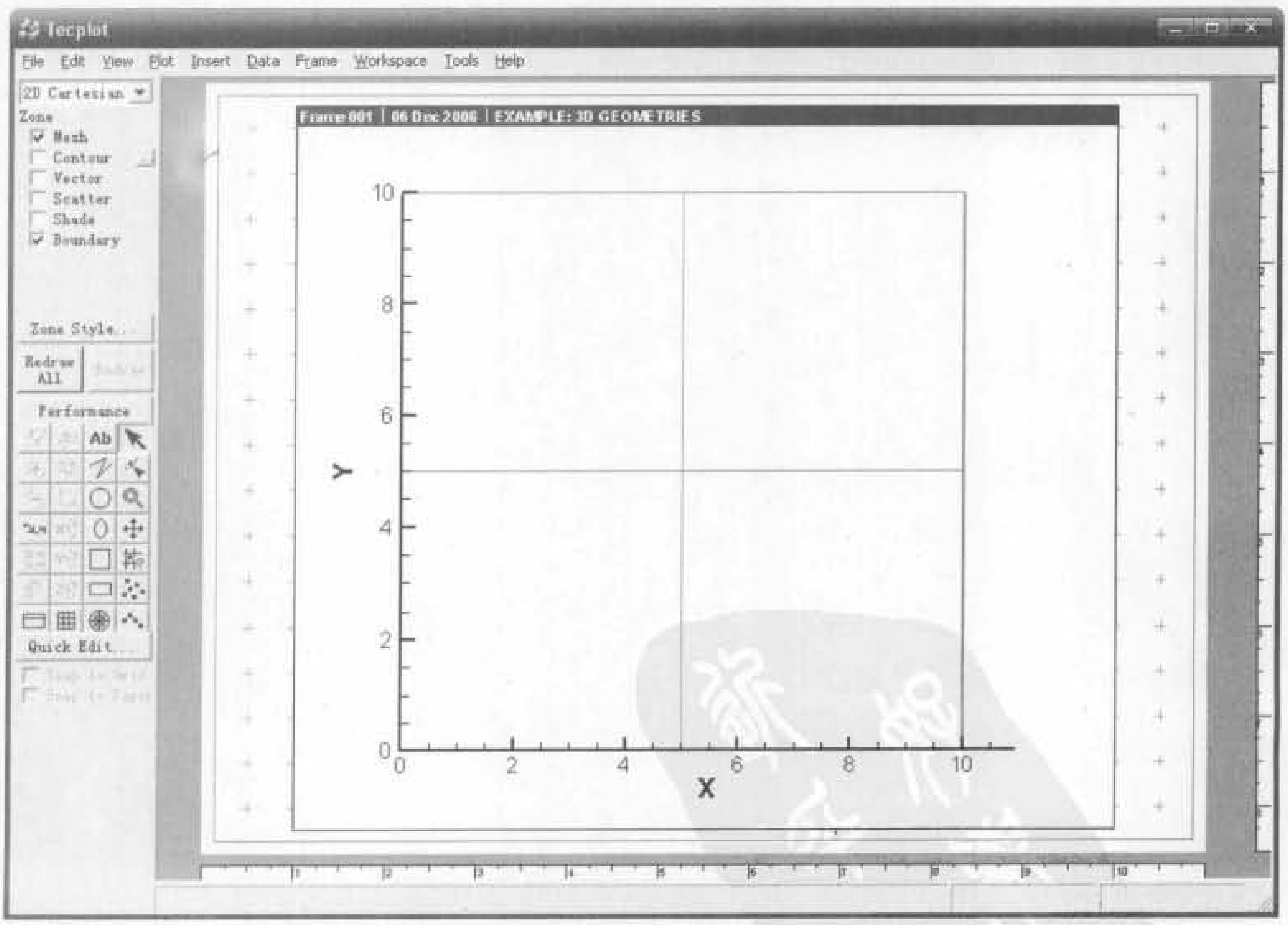


图 4-1　Tecplot 的主界面

1. 菜单栏

菜单栏如图 4-2 所示，通过它可以使用绝大多数 Tecplot 的功能，它的使用方式类似于

一般的 Windows 程序，是通过对话框，或者二级菜单来完成的。



图 4-2　菜单栏

Tecplot 菜单具体描述如下。

File：进行文件的读写、打印、输出曲线、记录宏、设定记录配置、退出。

Edit：实现剪切、复制、粘贴、清除、上提与下压显示顺序、修改数据点等功能。Tecplot 的剪切、复制和粘贴只在Tecplot内部有用，如果想和Windows的其他程序交换图形，可以用copy plot to clipboard 功能。

View：用来控制观察数据位置，包括比例、范围、三维旋转，还可以用来进行帧之间的粘贴。

Plot：控制轴的设置、区域类型、流线、云图(等值线或面)、矢量、离散点、颜色设置等功能。

Insert：控制文本、几何体(多线、圆、椭圆、正方形、矩形)、图片等功能。

Data：用来创建、操纵、检查数据。在 Tecplot 中可以进行的数据操作包括创建区域、插值、三角测量以及创建和修改由类似 FORTRAN 公式创建的数据。

Frame：创建、编辑、控制帧。

Workspace：用来控制工作区的属性，包括色彩图例、页面网格、显示选项和标尺。

Tools：用来快速运行宏，可以定义、创建或者创建简单的动画。

Help：打开帮助文档。

2. 工具栏

通过 Tecplot 的工具栏，可以进行经常用到的画图控制。许多工具的外形类似于要进行工作的性质。另外还可以控制帧的模式、活动帧和快照模式。工具栏如图 4-3 所示。

1)　帧模式

帧模式决定了当前帧显示的图形格式，如图4-4所示，共有4种。

- 3D：创建三维面或者体图像。
- 2D：创建二维图。
- XY：XY 曲线图。
- S(草图)：没有数据的图形，如流动图表和视图。

2)　区域/图形层

该选项决定了帧显示数据的格式。完全的绘图内容包括所有的图层、文字、几何形状，以及添加于图形基本数据其他内容。共有 6 种区域的二维和三维帧模式，4 种 XY 帧模式，但没有草图模式。

6 种二维和三维区域帧模式如图 4-3 所示，具体描述如下。

- Mesh(网格)：网格区域层用线连接数据点。
- Contour(等值线)：等值线区域层绘制等值线，可以是线或者常值或线间的区域，或者两者都有。



图 4-3 工具栏

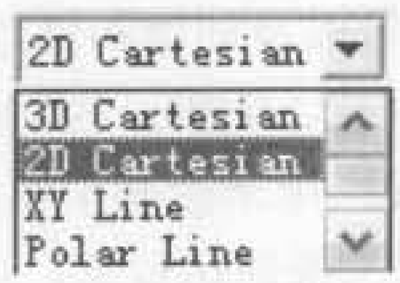


图 4-4 帧模式

- Vector(矢量)：绘制数值方向与大小。
- Scatter(散点)：在每一个数据点绘制符号。
- Shade(阴影)：用指定的固体颜色对指定区域进行着色，或者对三维绘图添加光源。
- Boundary(边界)：对于指定区域绘制边界。

4种XY模式下的图形层如图4-5所示。

- Lines(线状图)：用线状图绘制一对变量，线可以为分段线或者逼近线。
- Symbols(符号图)：用符号图绘制一对变量，由一个符号代表独立的数据点。
- Error Bars(误差柱状图)：将一对变量绘制成带误差的柱状图，可以用几种格式绘制误差柱状图。
- Bars(柱状图)：用柱状图绘制一对变量，可以是水平或者垂直图表。

3) 区域效果

对于三维帧来说，会出现如图4-6所示的复选框。只对着色的等值线绘图起作用。

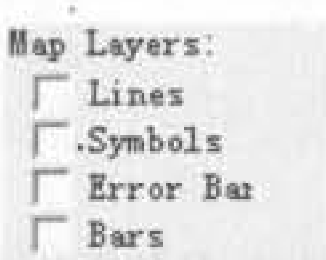


图 4-5 XY 图的形态

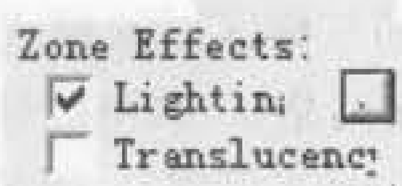


图 4-6 区域效果选项

4)　重画按钮

Tecplot 并不在每次图表更新后都自动重画，除非选择自动重画(automatically redraw)，还可以用 Redraw 按钮手动更新。

- Redraw(重画)：指重画当前帧。
- Redraw All(全部重画)：重画全部帧，按 Shift 键并单击 Redraw All 按钮会重新生成工作区。

自动重画(auto redraw)会连续不断地自动更新图表。

5)　显示选项按钮(display option button)

用来设定 Tecplot 的状态栏和性能参数。

6)　绘图属性按钮(plot attributes button)

可以打开绘图属性对话框进行区域显示设置。

7)　工具按钮(tool button)

每一个工具按钮都有相应的鼠标指针形状。共有 28 种，12 类，如图 4-7 所示。



图 4-7　工具栏与鼠标模式

- 等值线鼠标模式：有 3 个按钮，选中在线型显示模式下标注水平值，在 contour(云图)中增加水平，在 contour 中减少水平。
- 流线鼠标模式：有两个按钮，一次绘制一条流线，创建流线终止线。
- 帧鼠标模式：创建一个新的帧。
- 创建域鼠标模式：创建一个矩形域，创建一个圆形域。
- 三维旋转鼠标模式：6 个按钮，如图 4-8 所示。
- 文字鼠标模式：Ab创建文字标注。
- 几何形状鼠标模式：5 个按钮，如图 4-9 所示。可以在帧内进行几何体的草绘。



图 4-8　三维旋转按钮



图 4-9　草绘按钮

- 鼠标指示模式：进行选择与调整。
- 视图模式：进行放大、移动等视图显示操作。
- 鼠标探测模式：显示鼠标所在位置的物理量的值。
- 导出数据鼠标模式：如图 4-10 所示的两个按钮，进行数据的导出操作。

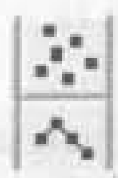


图 4-10　数据导出按钮

3. 状态栏

状态栏位于 Tecplot 窗口底部，在鼠标指针移过工具栏时会给出帮助提示。工具栏可以在 File→Preferences 中设定。

4. 工作区

工作区如图 4-11 所示，是进行绘图工作的区域。绘图工作都是在帧中完成的，类似于操作一个窗口。在默认情况下，Tecplot 显示网格和标尺。所有的操作都是在当前帧中完成的。

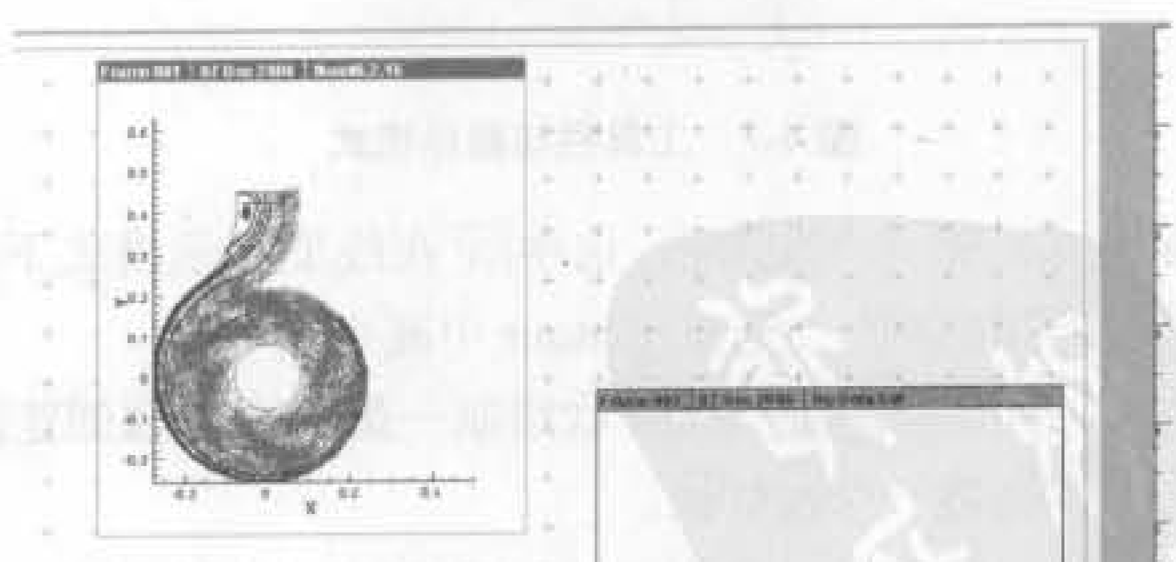


图 4-11　工作区

4.1.2　基本功能

Tecplot 是一种数据后处理软件，可以处理多种数据，并以多种形式表现出来，从简单的 XY 图到复杂的三维动态模拟，Tecplot 可快捷地将大量的资料转成容易了解的图表及图形。表现方式有等高线、三维流线、网格、向量、剖面、切片、阴影、上色等。

主要有以下功能。

- 可直接读入常见的网格、CAD 图形及 CFD 软件(PHOENICS、Fluent、STAR-CD)生成的文件。
- 能直接导入 CGNS、DXF、Excel、GRIDGEN、PLOT3D 格式的文件。
- 能导出的文件格式包括 BMP、AVI、Flash、JPEG、Windows 等常用格式。
- 能直接将结果在互联网上发布，利用 FTP 或 HTTP 对文件进行修改、编辑等操作。也可以直接打印图形，并在 Microsoft Office 上复制和粘贴。
- 可在 Windows 9x/NT/2000/XP 和 UNIX 操作系统上运行，文件能在不同的操作平台上相互交换。
- 利用鼠标直接单击即可知道流场中任一点的数值，能随意增加和删除指定的等值线(面)。
- ADK 功能使用户可以利用 FORTRAN、C、C++等语言开发特殊功能。

4.2　Tecplot 的数据格式

4.2.1　Tecplot 数据层次

Tecplot 中的数据分为两个层次，如图 4-12 所示。

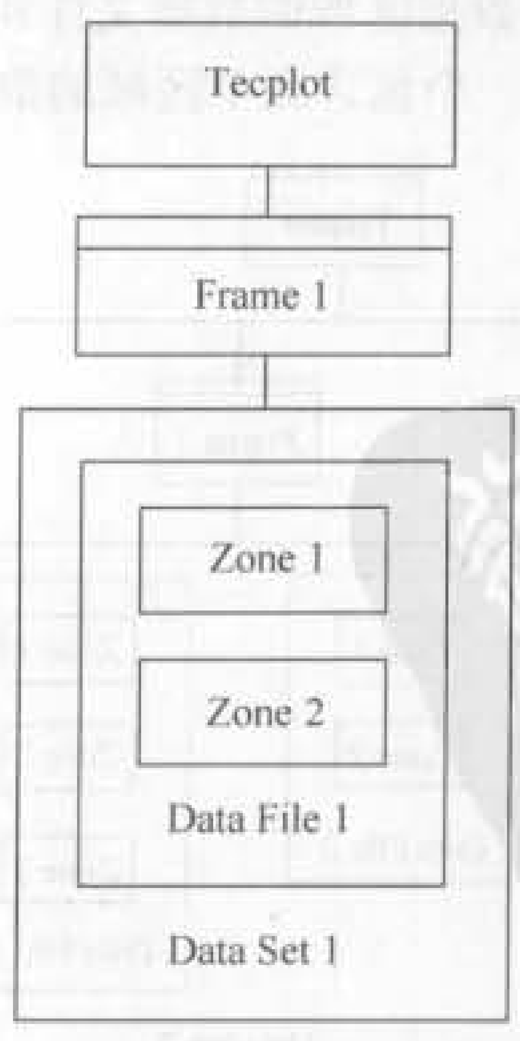


图 4-12　Tecplot 数据结构示意图

Tecplot 中最高等级的数据被称作一个数据系列(Data Set)。它包括一个或者多个数据区

域(Zone)、数据块等。区域为数据结构中的第二等级，可以从数据文件中读入或者利用 Tecplot 进行创建。在运行 Tecplot 软件时，每当读入数据文件，或者创建区域时系统便会把数据加入到活动帧的数据结构中去。同一个数据系列可以和多个帧(Frame)连接，如图 4-13 所示。

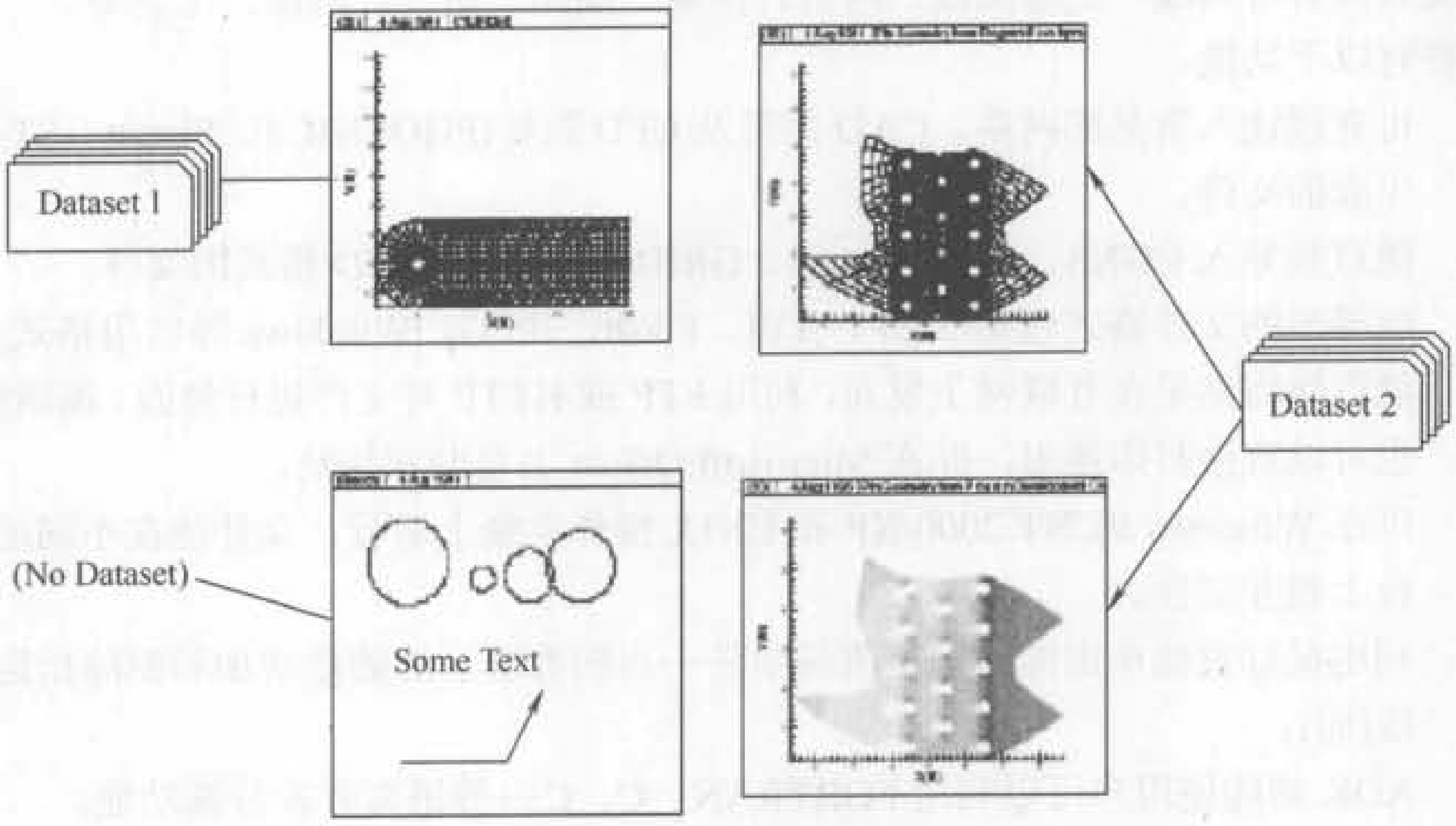


图 4-13　多帧数据连接方式

如果读入的数据文件超过一个，Tecplot 软件会自动将数据分组为一个数据系列，而且对每个数据点都包含有相同的变量参数，但是并不要求所有的数据文件的参数顺序都相同。Tecplot 中应用相同数据系列帧的标题颜色相同。图 4-14 给出了一个复杂的 Tecplot 数据文件结构。图中，帧 1 和帧 2 利用数据系列 1，由一个包含 3 个区域的数据文件构成；帧 3 利用数据系列 2，由一个包含两个数据区域的数据文件和一个包含 3 个数据区域的数据文件组成；帧 4 利用数据系列 3，由一个包含两个区域的数据文件构成。

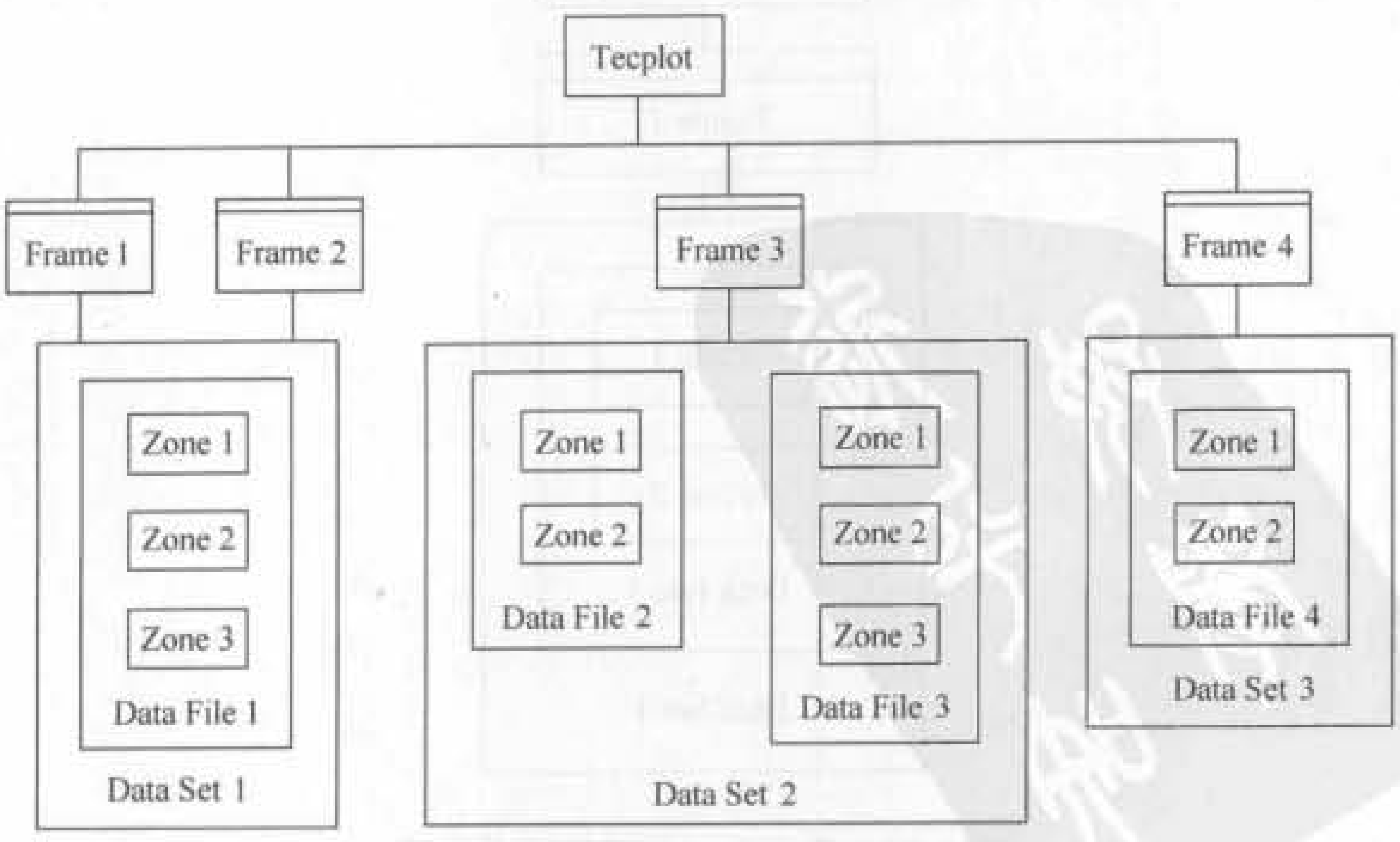


图 4-14　复杂的 Tecplot 数据文件结构

4.2.2　多数据区域

多数据区域可以用来方便绘制复杂结构，或者细分结构图标；也可以用来表示不同时间步的数据，或者不同测量方法的数据。图 4-15 所示为一个利用多数据区域的范例。

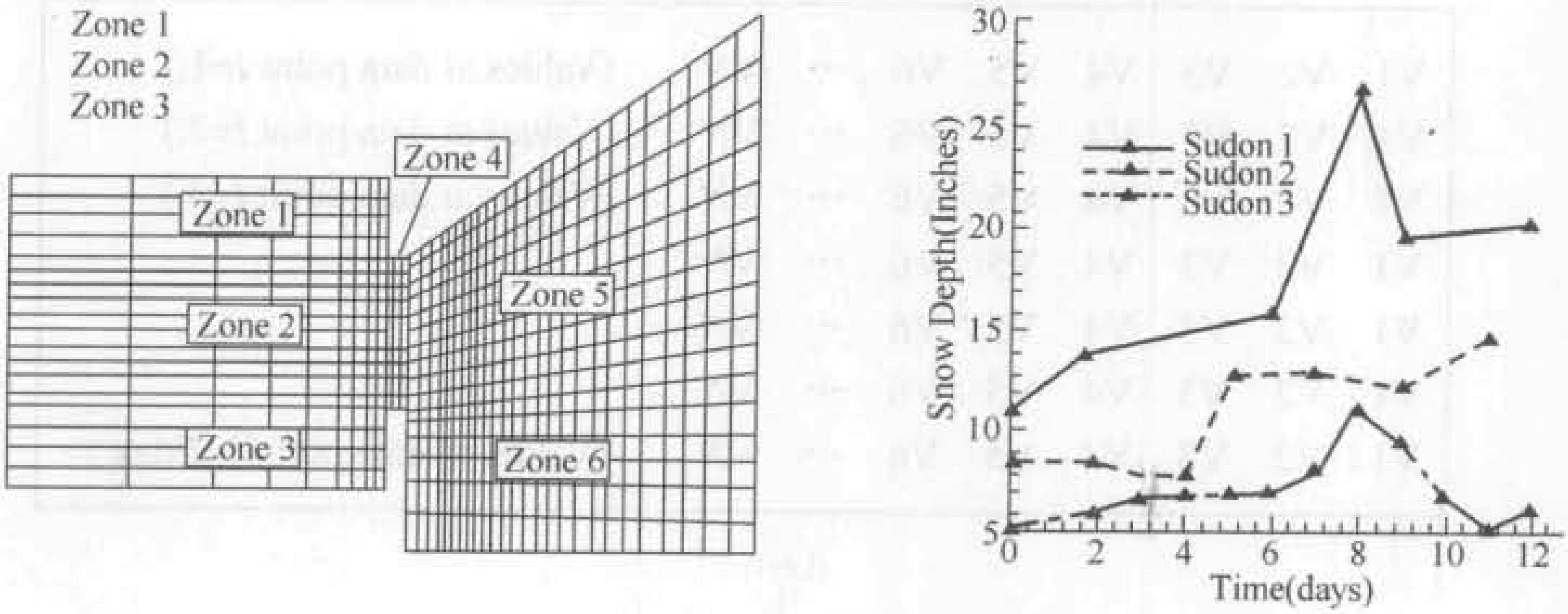


图 4-15　多数据区域应用示例

4.2.3　数据区域中的数据结构

Tecplot 可以使用两种数据类型：有序数据和有限元数据。

1. 有序数据

有序数据是一列按照逻辑保存于一维、二维或者三维数据组中的数据。在Tecplot中，I、J、K用来表示数据组维数下标。最常见的数据形式有如下几种。

- I 序列：I 维数据组点数大于 1 并且 JK 维数据点数为 1。I 维数据点数为整个数据组数据点数。
- IJ 序列：二维数组 IJ 的数据点数大于 1 并且 K 维数据点数为 1，数据点数为 IJ 维数据点的乘积。
- IJK 序列：三维数据组中 IJK 维数据点数大于 1，数据点个数为 IJK 数据点个数的乘积。

其他系列的数据格式可能也有效，但却不能用 Tecplot 进行创建，但可以由其他的程序进行创建，例如下面的几个例子。

- J 序列或者 K 序列：类似于 I 系列，却变成了 J 或者 K 维数据点数大于 1 并且其他维数据点个数等于 1。
- JK 和 IK 序列：类似于 IJ 序列，都是三维中两维数据点数大于 1，而剩余 1 维数据点数为 1。

1)　I、J 或者 K 序列数据

通常有一维数组，由某个参数作为指数。对于 I 序列，最常见的形式为 I=1,I=2,I=3,…,$I=I_{max}$，如果用表格来表示，共有 N 个变量(V1,V2,V3,…,VN)。每一行代表一个参数。如表 4-1 所示。

2) IJ 序列数据

可以应用于二维或者三维表面场曲线，按照两参数网格组织。每一个数据点有一系列的两个参数(I和J)，并且具有4个相邻数据点，如图4-16和图4-17所示。

表 4-1 I 序列数据表(适用于 XY 曲线绘图)

V1	V2	V3	V4	V5	V6	···	V*N*	(Values at data point *I*=1,)
V1	V2	V3	V4	V5	V6	···	V*N*	(Values at data point *I*=2,)
V1	V2	V3	V4	V5	V6	···	V*N*	(Values at data point *I*=3,)
V1	V2	V3	V4	V5	V6	···	V*N*	
V1	V2	V3	V4	V5	V6	···	V*N*	
V1	V2	V3	V4	V5	V6	···	V*N*	
V1	V2	V3	V4	V5	V6	···	V*N*	(Values at data point *I*=*I*Max,)

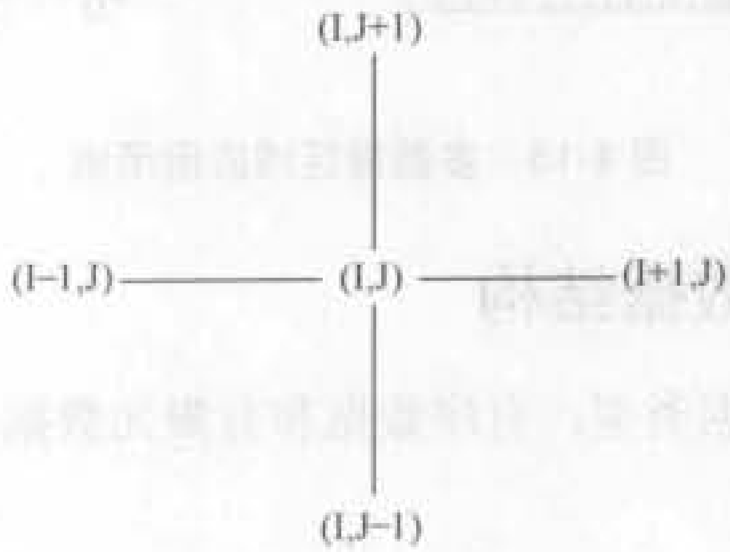


图 4-16 IJ 系列数据点相邻数据

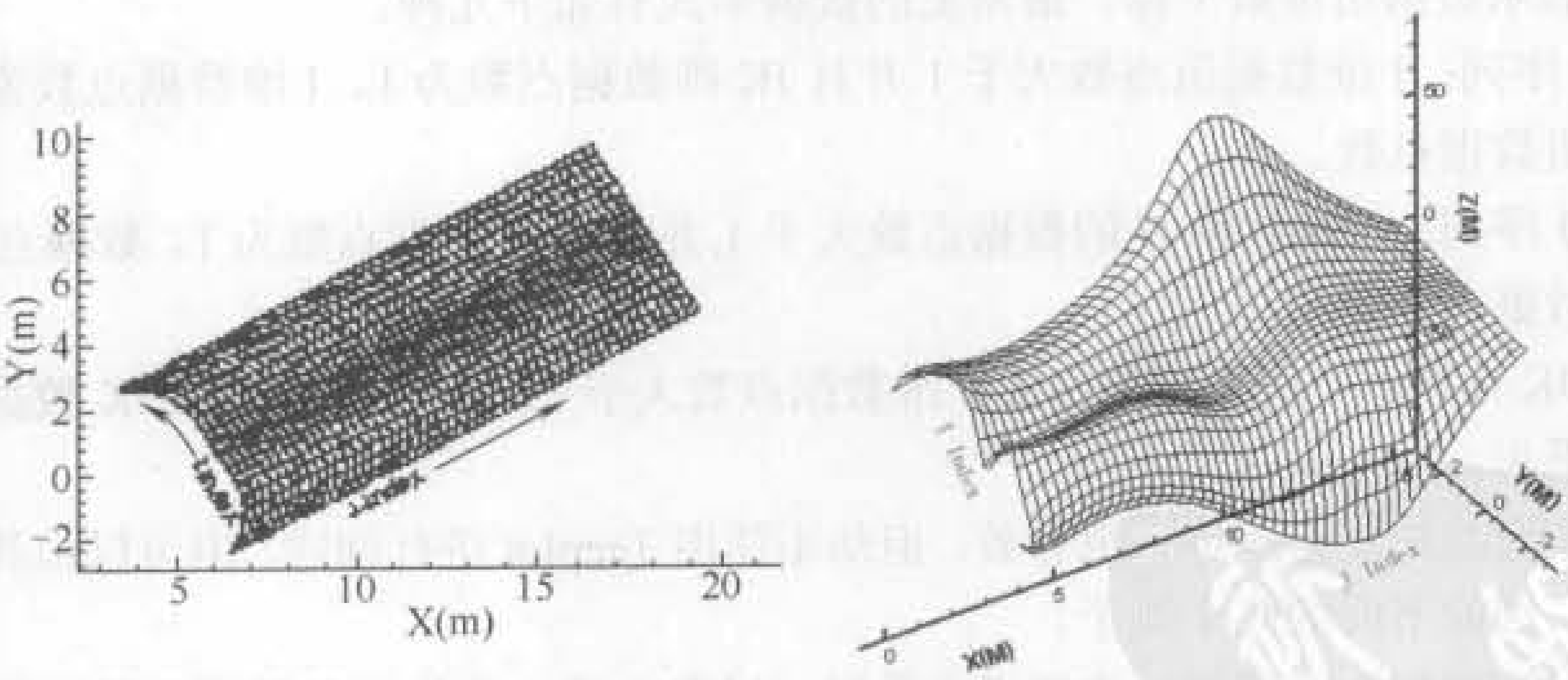


图 4-17 左为二维帧模式，IJ 序列数据点；右为三维帧模式，IJ 序列数据点

3) IJK 序列数据点

IJK 序列数据点通常按照三参数网格组织。每个点有一系列的 3 个参数(I,J,K)，并且有 6 个相邻数据点(边界除外)，如图 4-18 所示。

4) I 平面、J 平面和 K 平面

对于 IJK 序列数据来说，一个很重要的概念便是 I 平面、J 平面和 K 平面。K 平面是按照有相同 K 指数的数据点相连而成的平面，I、J 用来表示在整个区域中的范围。这样，

K 平面实际上有两个参数顺序，类似于 IJ 顺序。注意，K 平面并不一定是严格意义上的平面，这不过是代表它存在于 IJK 空间。简单示例如图 4-19～图 4-22 所示。

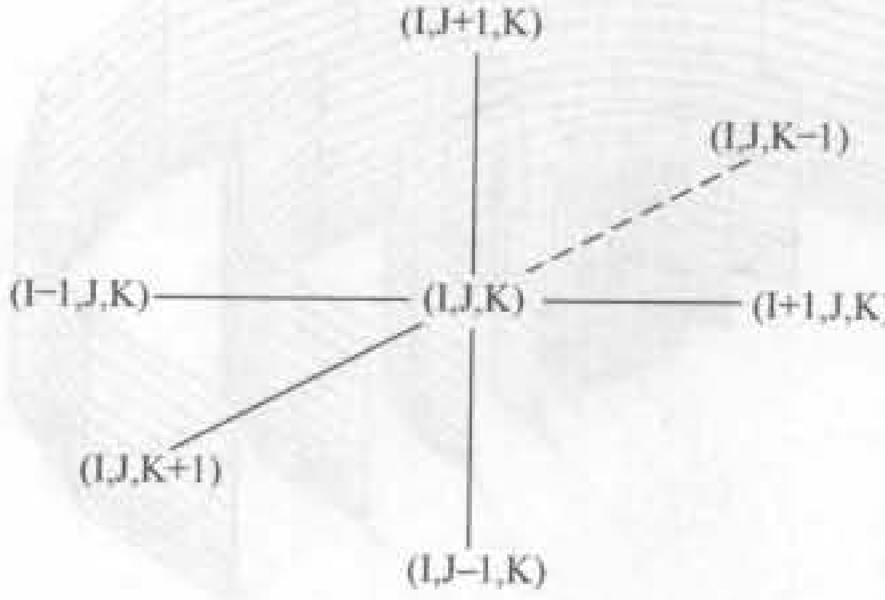


图 4-18　IJK 序列数据点相邻数据

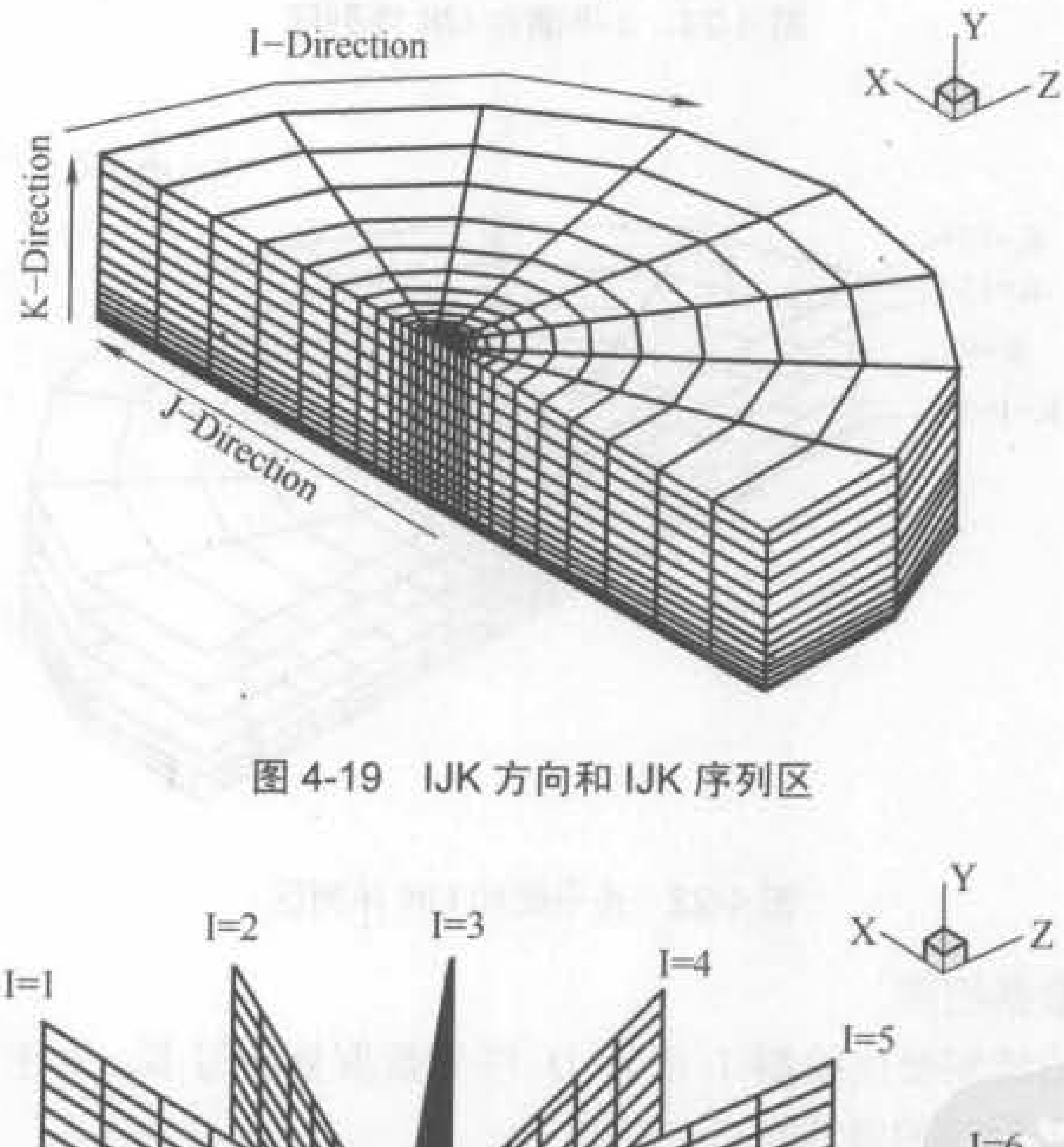


图 4-19　IJK 方向和 IJK 序列区



图 4-20　I 平面和 IJK 区

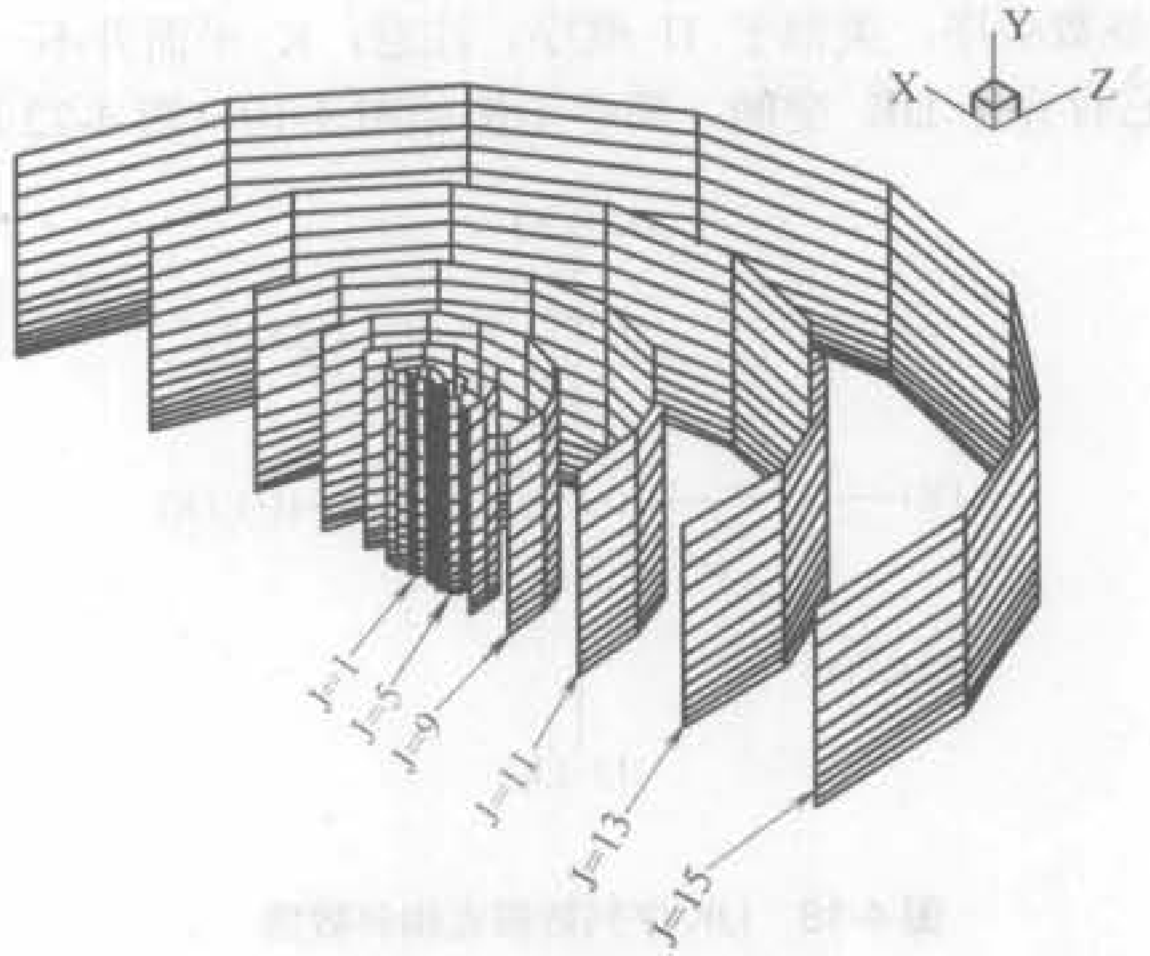


图 4-21 J 平面和 IJK 序列区

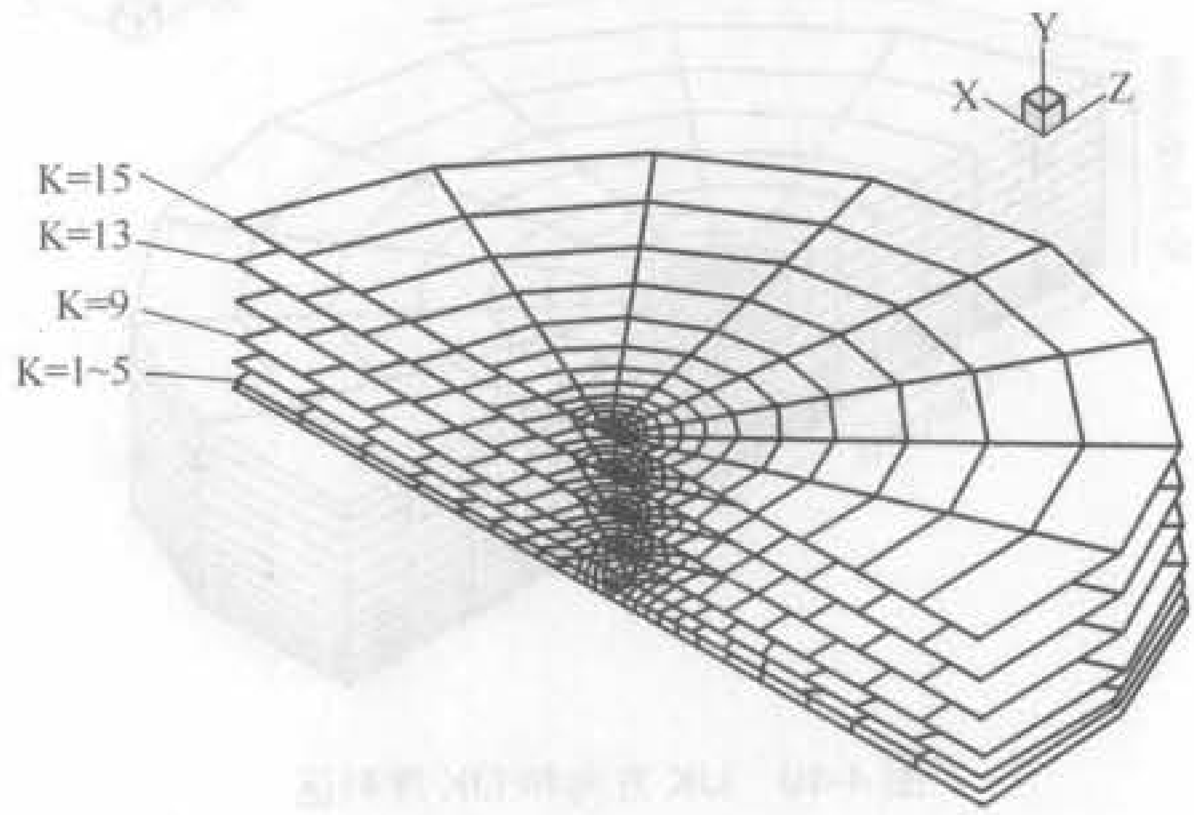


图 4-22 K 平面和 IJK 序列区

5) IJK 序列数据绘图

IJK 序列数据的绘制要比绘制 I 或者 IJ 序列数据复杂得多，对于 IJK 数据可以有更多的选择来确定显示数据的方式。

2. 有限元数据

有限元(finite-elementary)数据或称 FE 数据，是一种把数据点作为二维或者三维空间中的点按照规定连接形成单元或者网格的数据结构方法。

有限元数据可以分为两类。

- FE-表面：用系列三角形或者四边形定义二维场或者三维面。
- FE-体：用系列四面体或者块单元定义三维场。

对于任何一种类型，理论上来说没有数据点个数的极限，但是由于计算机内存容量的限制实际上是有极限的。

1)　FE-体数据

有限元体数据可以包含 4 个点(四面体)或者 8 个点(块)。每个区域的单元数据必须全部为四面体或者块，不能同区中同时包含两种数据。有限元体的节点连接方式如图 4-23 和图 4-24 所示。在块格式中，点可以重复连接 4—点元，5—点元，6—点元和 7—点元。例如，节点列表如 n1n1n1n1n5n6n7n8，会生成四边形为底的棱锥单元。图 4-25 所示为一个有限元体数据绘图示例。

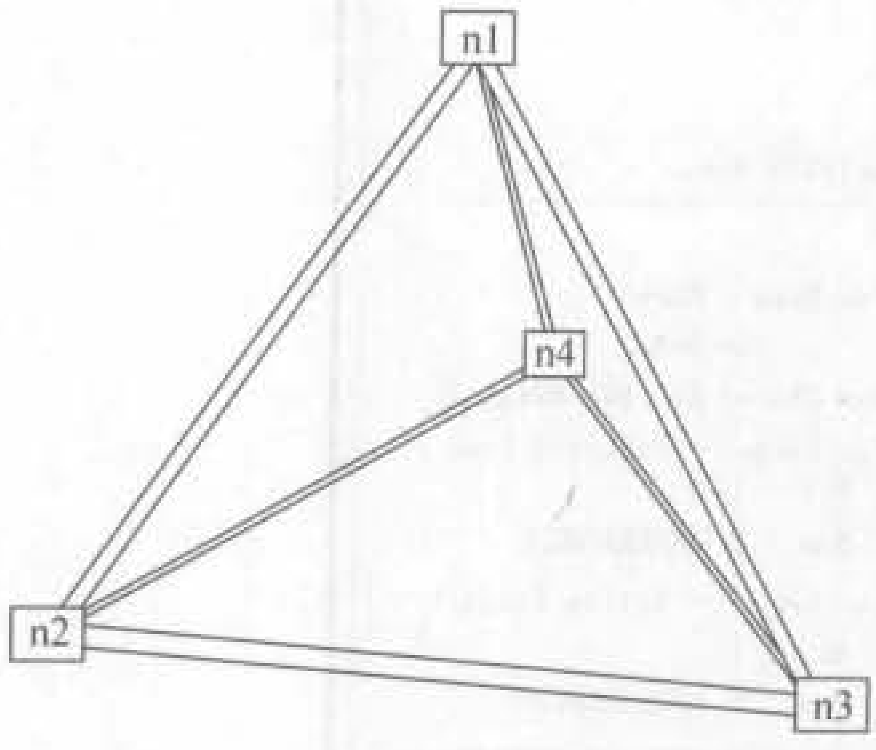


图 4-23　四面体有限元体节点连接方式

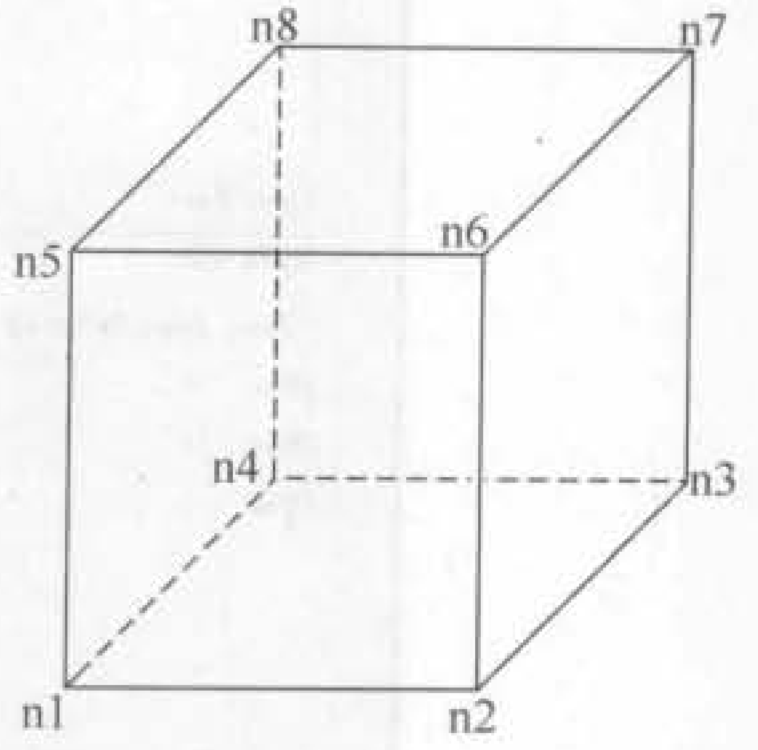


图 4-24　块有限元体连接方式

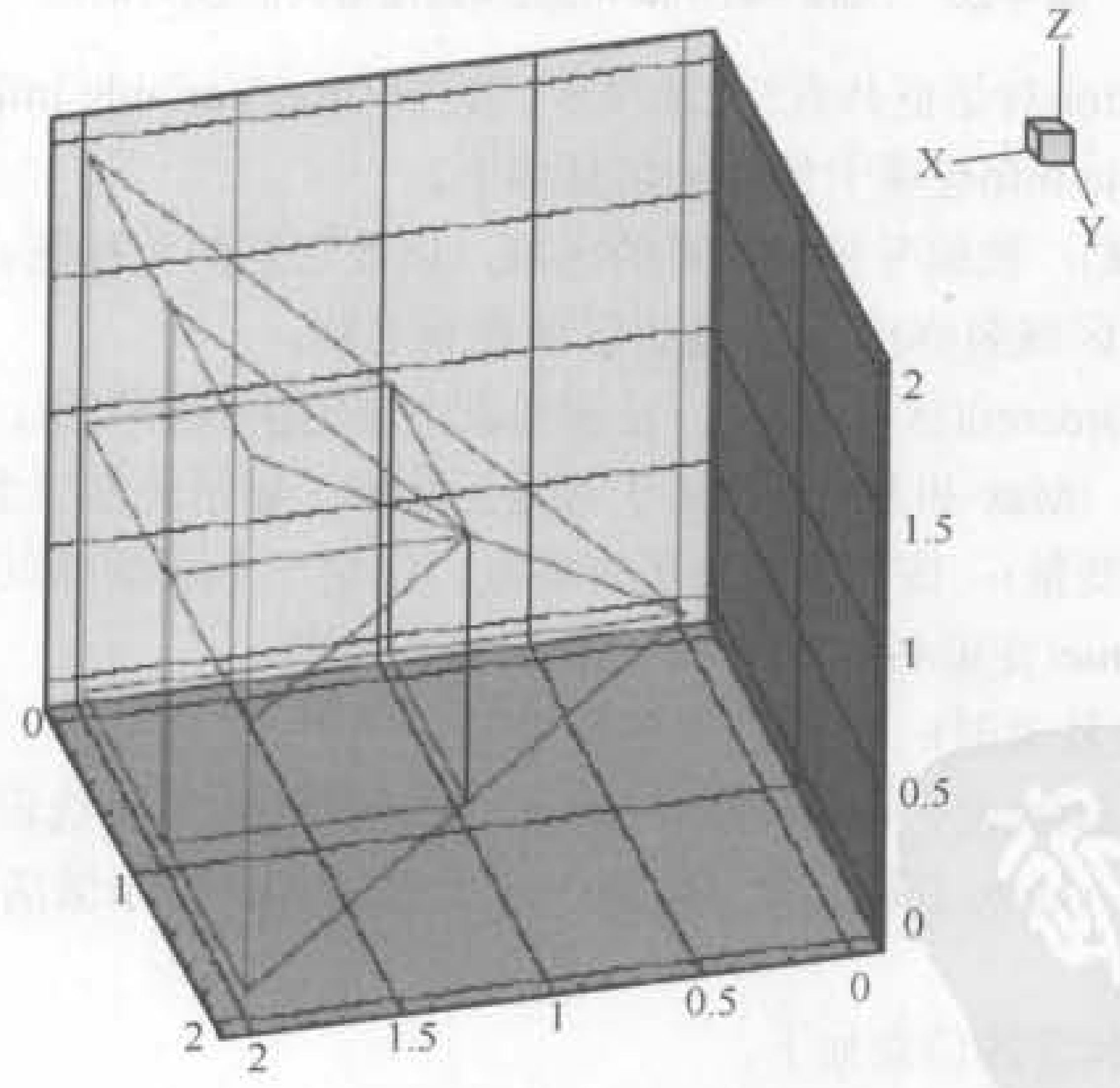


图 4-25　有限元件数据绘图示例

2)　查看数据系列信息

可以选择 Data(数据)→Data Set Info(数据系列信息)命令，来打开如图 4-26 所示的 Data Set Information(数据系列信息)对话框查看当前数据系列的信息，包括数据系列头、区域和变量名称，以及选择变量的最大值与最小值，也可以用来改变数据系列的头标识、区域、

变量名称等。

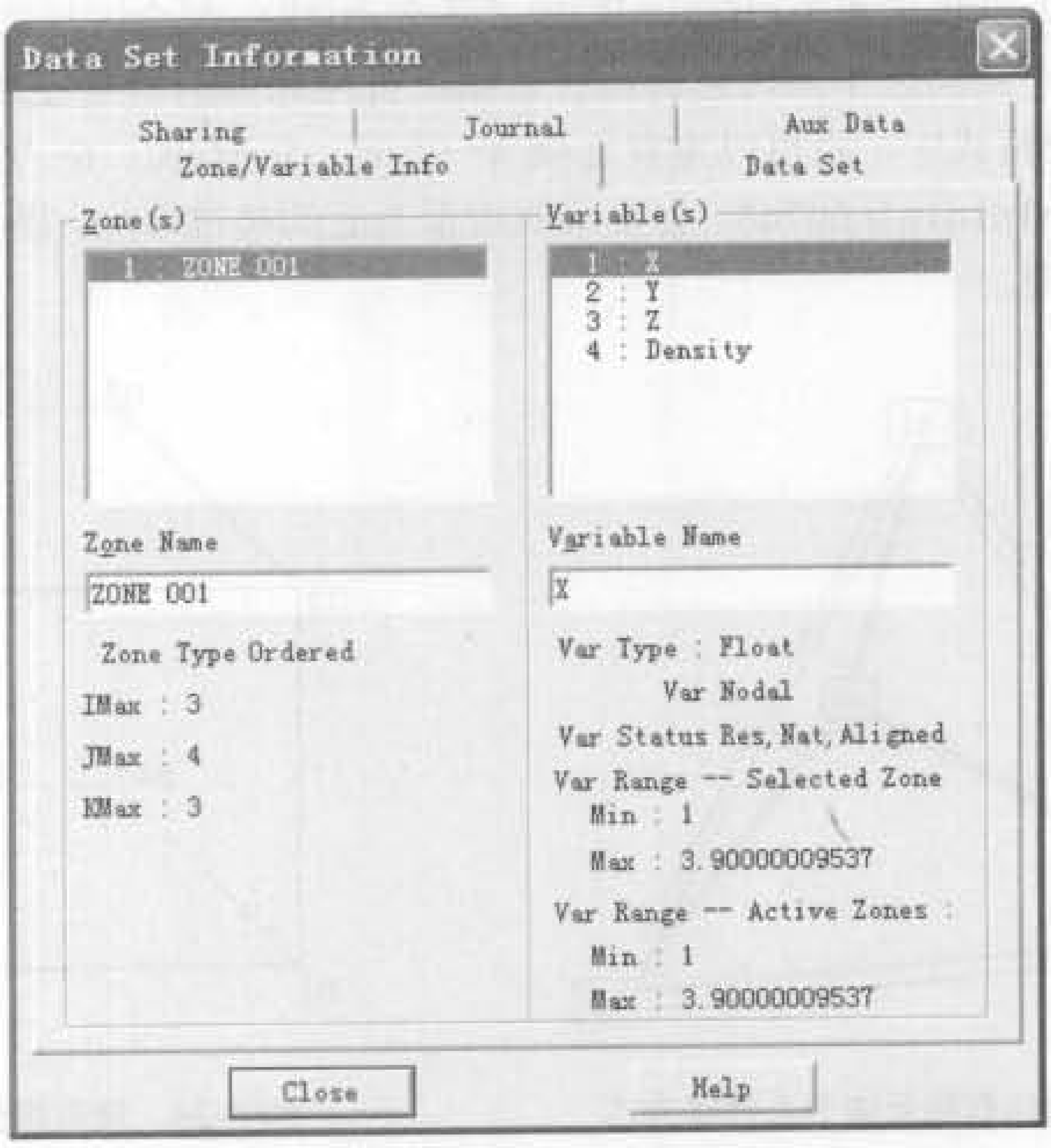


图 4-26 Data Set Information(数据系列信息)对话框

Data Set Information对话框共有5个选项卡，其中Zone/Variable Info和Data Set选项卡比较常用。Zone/Variable Info选项卡包含的信息如下。

- Zone(s)(区域)：按编号列出所有的区域，以及它们的头标题。
- Zone Name(区域名称)：输入选定区域的新名称。
- Zone Type Ordered(区域类型)：显示列表中选定区域的类型。对于 FE 序列数据，包含 IMax、JMax 和 KMax。对于有限元数据，显示元素类型，点数和单元个数。
- Variable(s)(变量)：按名称列出所有变量，选定一个变量可以更改其名称。
- Variable Name(变量名)：输入选定变量的新名称。
- Var Type(变量类型)：显示选定变量类型。
- Var Range--Selected Zone(变量范围——选定参数)：显示选定变量的取值范围。
- Var Range--Active Zones(变量范围——激活区域)：显示激活区域所有变量的最大值和最小值。

Data Set 选项卡包含的信息如下。

- Data Set Title(数据系列标题)：输入当前数据标题，或者编辑当前数据标题。
- Data File(数据文件)：列出所有外部数据文件的文件名和路径。
- Var Load Mode(变量加载模式)：根据所用的方法，该项为根据位置(by position)或者根据名称(by name)。

4.3 Tecplot 对 Fluent 数据进行后处理

4.3.1 Tecplot 识别的数据格式

Tecplot 可以直接读取十余种软件生成的数据文件，主要的描述如下。

- 计算流体力学通用注释系统格式(Computational fluid dynamics General Notation System，CGNS)。
- 数字评价图格式(Digital Elevation Mapformat，DEM)。
- 数字交换格式(Digital eXchange Format，DXF)。
- Excel 表(Windows only)。
- Fluent 5.0 以上(.cas and .dat)。
- Gridgen 格式。
- Hierarchical Data Format (HDF)。
- Image 文件。
- PLOT3D 文件。
- Text spreadsheet 文件。

Tecplot 选择读取数据文件的对话框如图 4-27 所示。

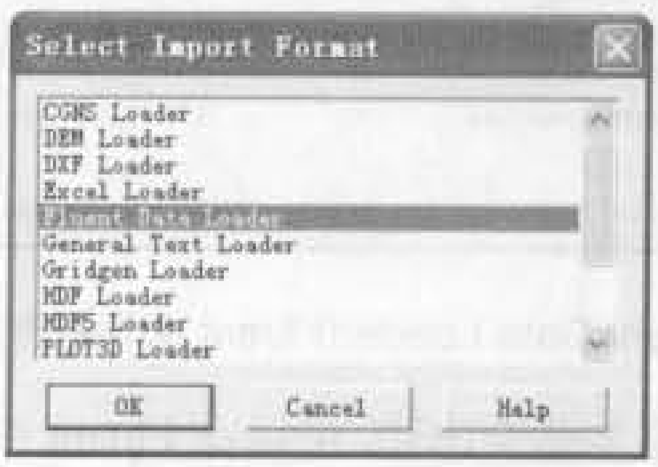


图 4-27 Select Import Format(选择导入格式)对话框

4.3.2 Tecplot 读取 Fluent 文件步骤

由于 Tecplot 只接受 Fluent 5.0 及以上版本的数据，所以要想读入早期版本的文件就必须首先读入 5.0 或以上版本重新保存后读入。Fluent 格式文件的数据都保存于网格中心。因为 Tecplot 要求所有的数据位于节点，所以在加载数据时会采用算术平均中心点数据从而给出节点数据。Fluent Data Loader(Fluent 文件数据加载器)对话框如图 4-28 所示。

该对话框中的主要选项介绍如下。

- Load Case and Data File(加载网格和数据文件)：读入*.cas 和*.dat 文件。Tecplot 会读入所有的变量并且加入 Tecplot 数据系列。
- Load Case File Only(只加载网格)：只读入 Fluent*.cas 文件。
- Load Residuals Only(只加载残差)：只加载残差数据(收敛记录)。
- Options(网格选项)：网格选项与要加载的*.cas 文件相联系，有如下选项。
- Case 文件：输入要加载的文件名或者从结果对话框中选择文件。

- ◆ Load Cells and Boundaries(加载单元和边界)：从*.cas 文件中读入单元数据和边界区域。每一个流体或者固体区域在 Tecplot 软件中会显示为单独的区域。
- ◆ Load Cells Only(只加载数据)：只加载单元数据区域。每个区域都作为独立 Tecplot 数据区域。
- ◆ Load Boundaries Only(只加载边界)：只加载边界区域，作为独立 Tecplot 数据区域。

- Data 文件：*.dat 文件包含求解数据和残差数据(收敛记录)，可以输入要加载的文件名称或者用结果对话框选择文件。

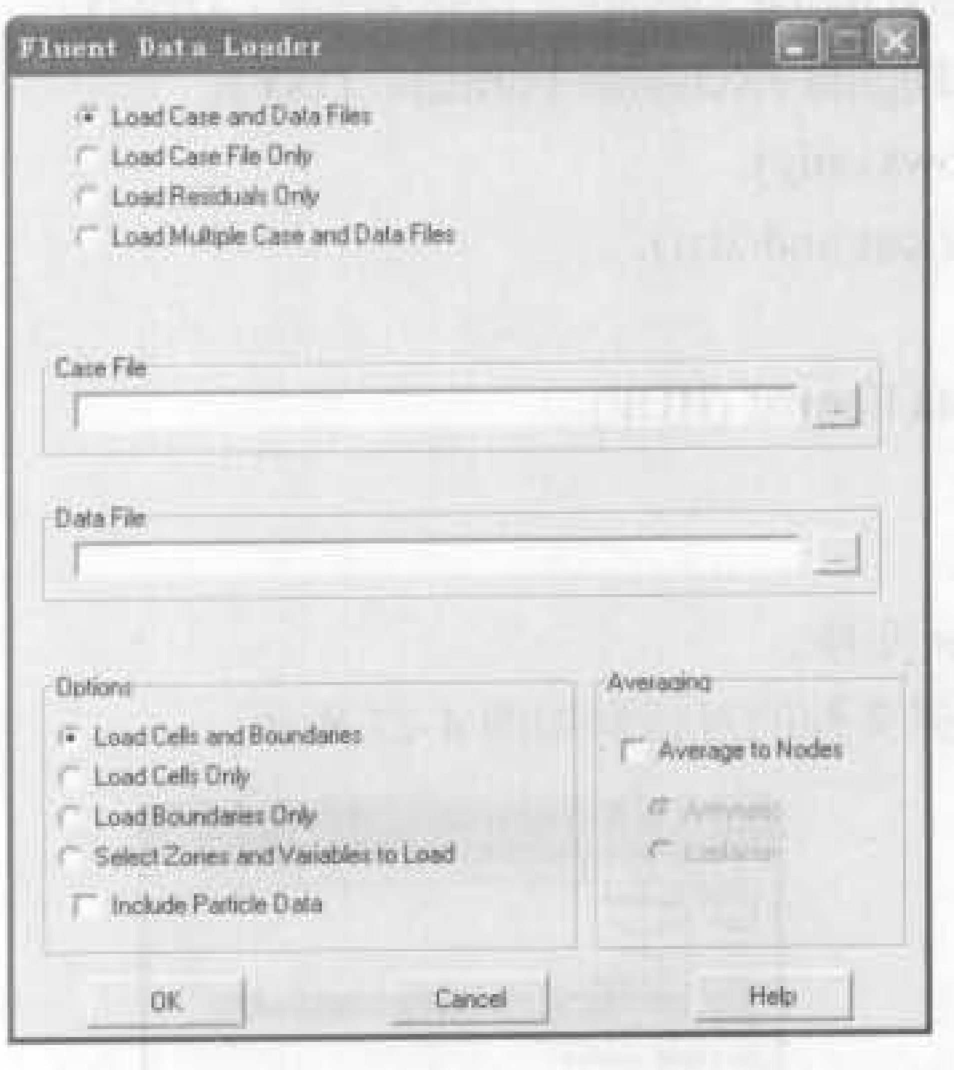


图 4-28 Fluent Data Loader(Fluent 文件数据加载器)对话框

对于有滑移网格 Fluent 数据加载会出错，可以在 Fluent 中写出感兴趣区域的 Tecplot 数据(选择 File→Export 命令，进入输出文件界面，如图 4-29 所示)，通过 Tecplot 读取数据文件进行后处理，其设置界面如图 4-30 所示。

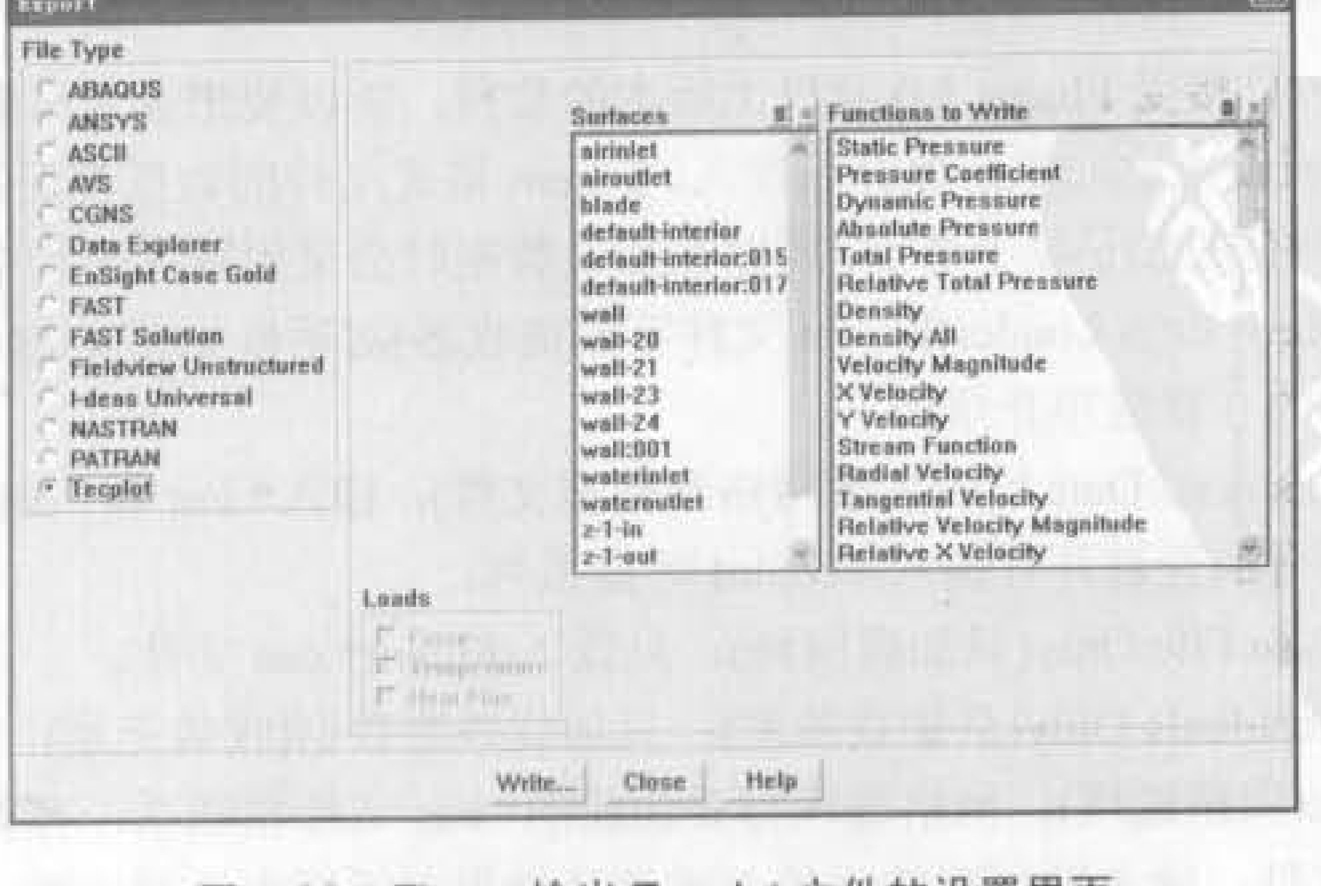


图 4-29 Fluent 输出 Tecplot 文件的设置界面

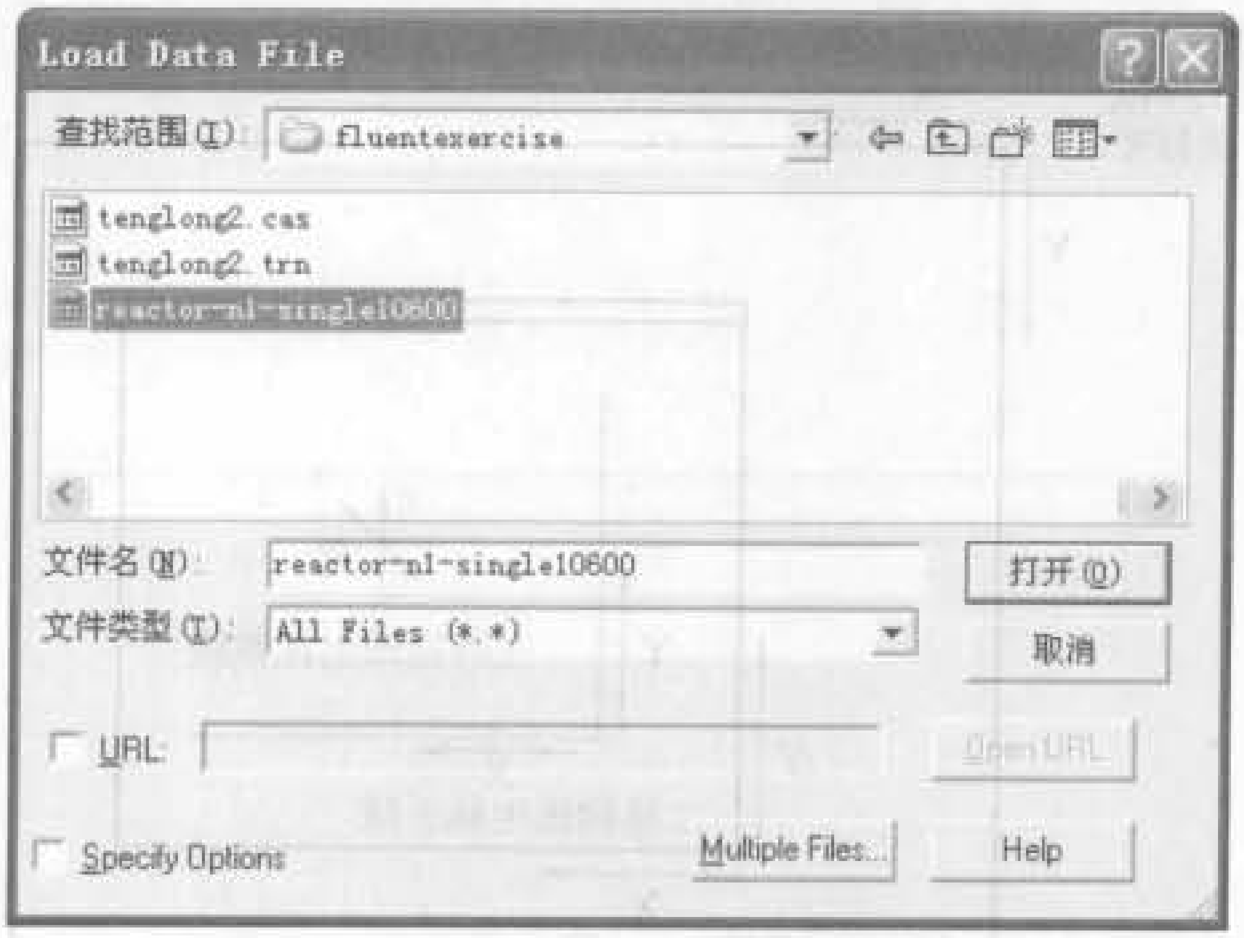


图 4-30　Load Data File(加载数据文件)对话框

4.4　Tecplot 绘图环境设置

工作区是进行所有的绘图操作的区域，通过对工作区域进行正确设置可以大大地方便绘图操作。

4.4.1　网格和标尺的设定

利用网格可以方便地定位对象，在添加文本和几何图形时可以选择对齐到网格。利用标尺可以方便地放大缩小对象。标尺的显示单位可以选择为厘米(cm)、英寸(in)、点数(pt)或者不显示标尺。要改变网格和标尺设定步骤如下。

(1) 选择 Workspace→Ruler/Grid 命令，打开 Ruler/Grid 对话框。

(2) 可以选择是否显示网格。

(3) 若显示网格，则在网格间距(Grid Spacing)下拉列表框中指定网格间距。

(4) 选择是否显示标尺。

(5) 若显示标尺，则在标尺间距(Ruler Spacing)下拉列表中选择合适的间距。

4.4.2　坐标系统

Tecplot 中包含多个坐标系统，但是工作纸、帧、2D 和 3D 坐标系统最为重要，如图 4-31 所示。

Tecplot 使用对象相对于帧的比例来确定对象的大小和位置。当在 Tecplot 中输入数值时可以使用不同的有效后缀，Tecplot 会自动转化为帧单位。例如，如果希望输入的文本位于对象上方 1 英寸，可以输入 1in，Tecplot 会自动进行单位转化。

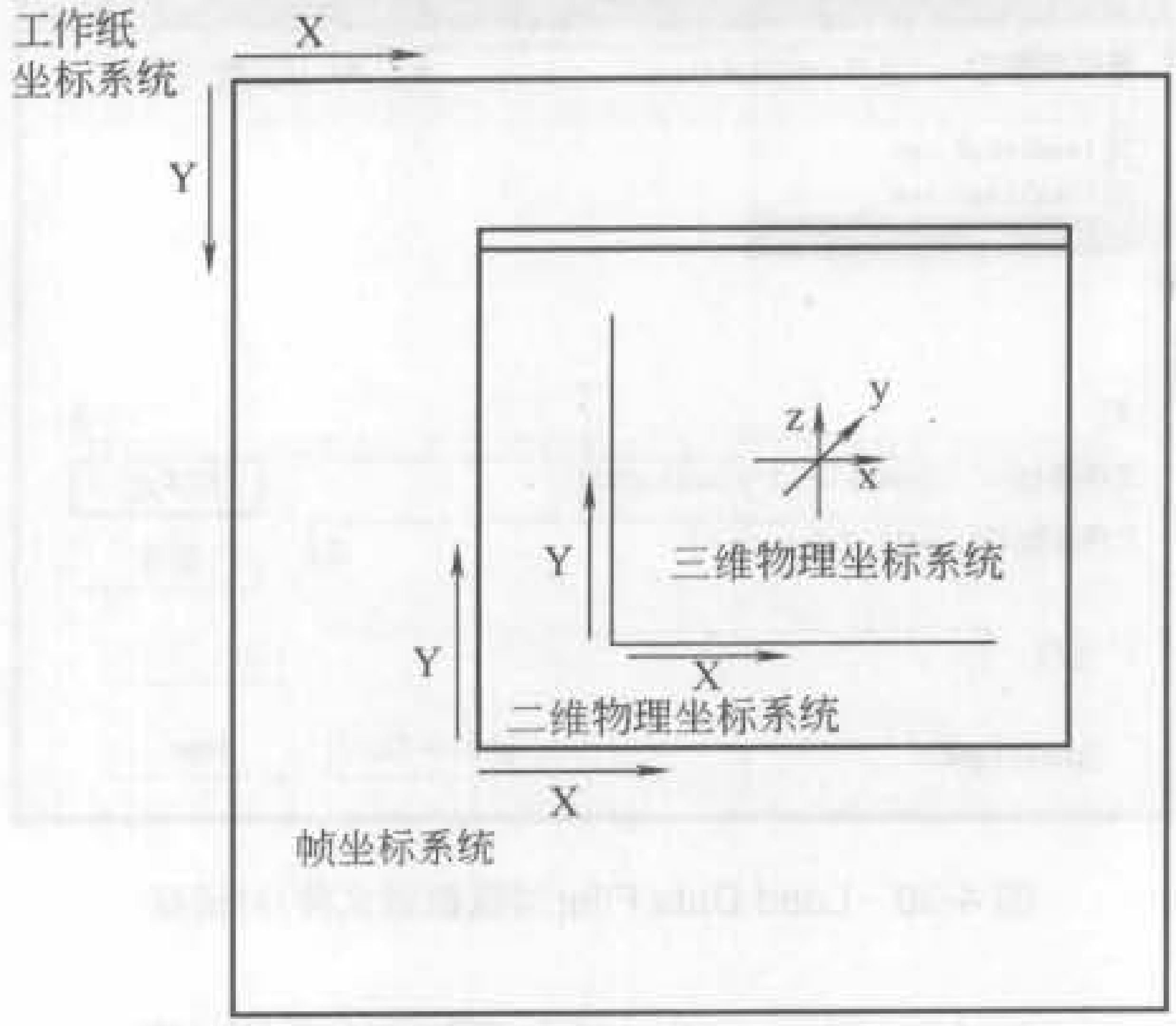


图 4-31 Tecplot 坐标系统

4.5 Tecplot 使用实例

4.5.1 绘制 XY 曲线

Tecplot 中的所有 XY 曲线都是由一个或者多个 XY 对构成的。XY 对之间的关系以及曲线绘制方式，在 Tecplot 中被统称为 XY 绘图。XY 绘图有三种方式。

- 直线式(Lines)：用线段连接所有的数据点。
- 符号式(Symbols)：每个数据点由一个符号代表，如圆、三角形、方形等。
- 柱状式(Bars)：每一个数据点由一个水平或垂直柱代表。

下面结合 Tecplot 自带的 rain.plt 文件讲解 XY 曲线的绘制。

(1) 选择 File→New Layout 命令。

(2) 选择 File→Load Data File 命令，在打开的对话框中选择 Tecplot 的安装目录 TEC100 下的“Demo\plt\rain.plt。

(3) 在 XY 菜单中，选择 Define XY Mappings(定义 XY 绘图)命令。选中 Map 2 和 Map 3，然后单击 Map Show 项会出现一个选择对话框，选择 Activate(激活)，如图 4-32 所示。

(4) 在 XY-Mappings 对话框中有许多选项，可以方便地用来执行改变曲线形式、定义 XY 轴变量、改变柱状图形式等功能，由于其操作过程非常简单，这里不再一一赘述。图 4-33 为做好的一个示例图。

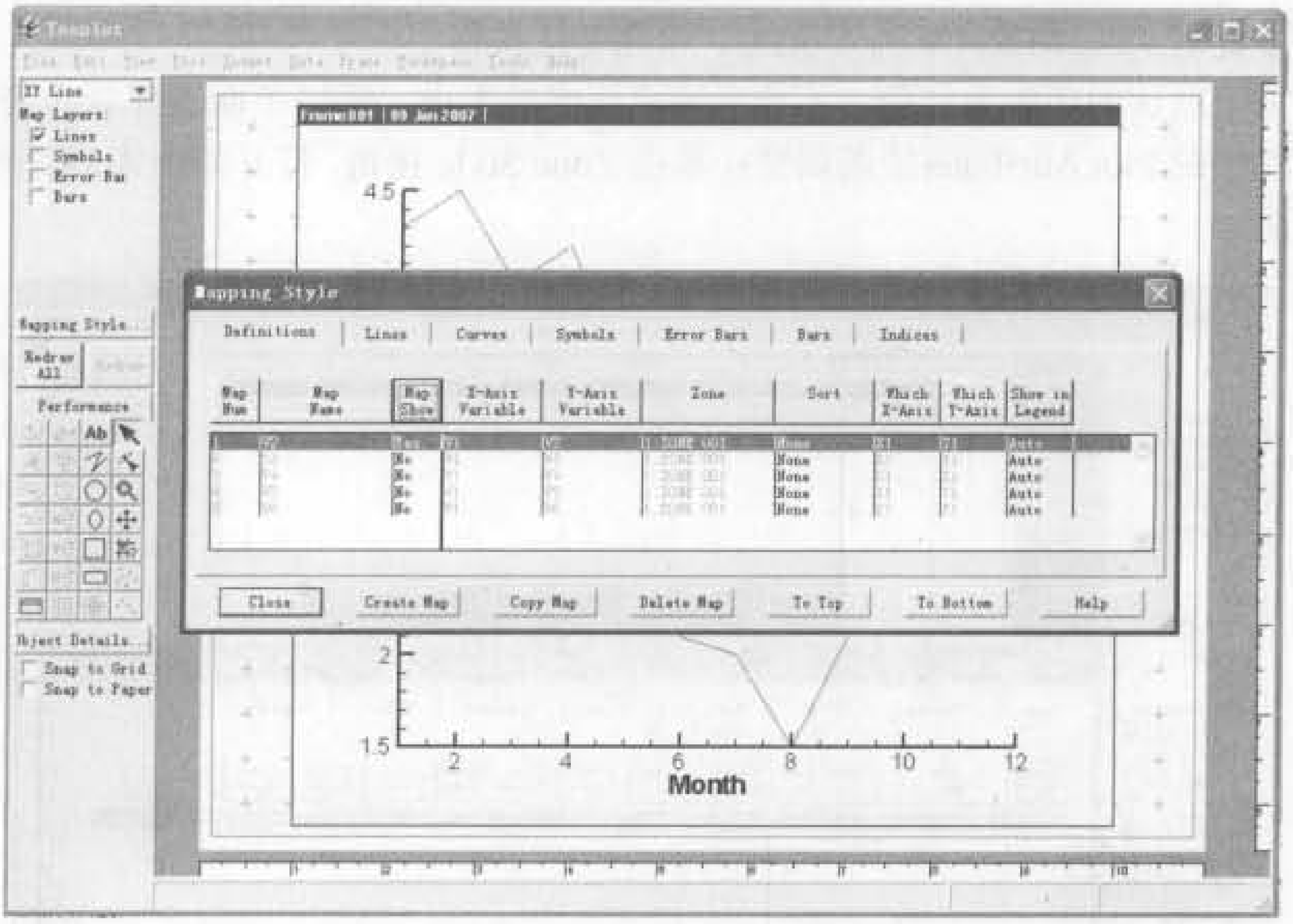


图 4-32　Mapping Style 设置

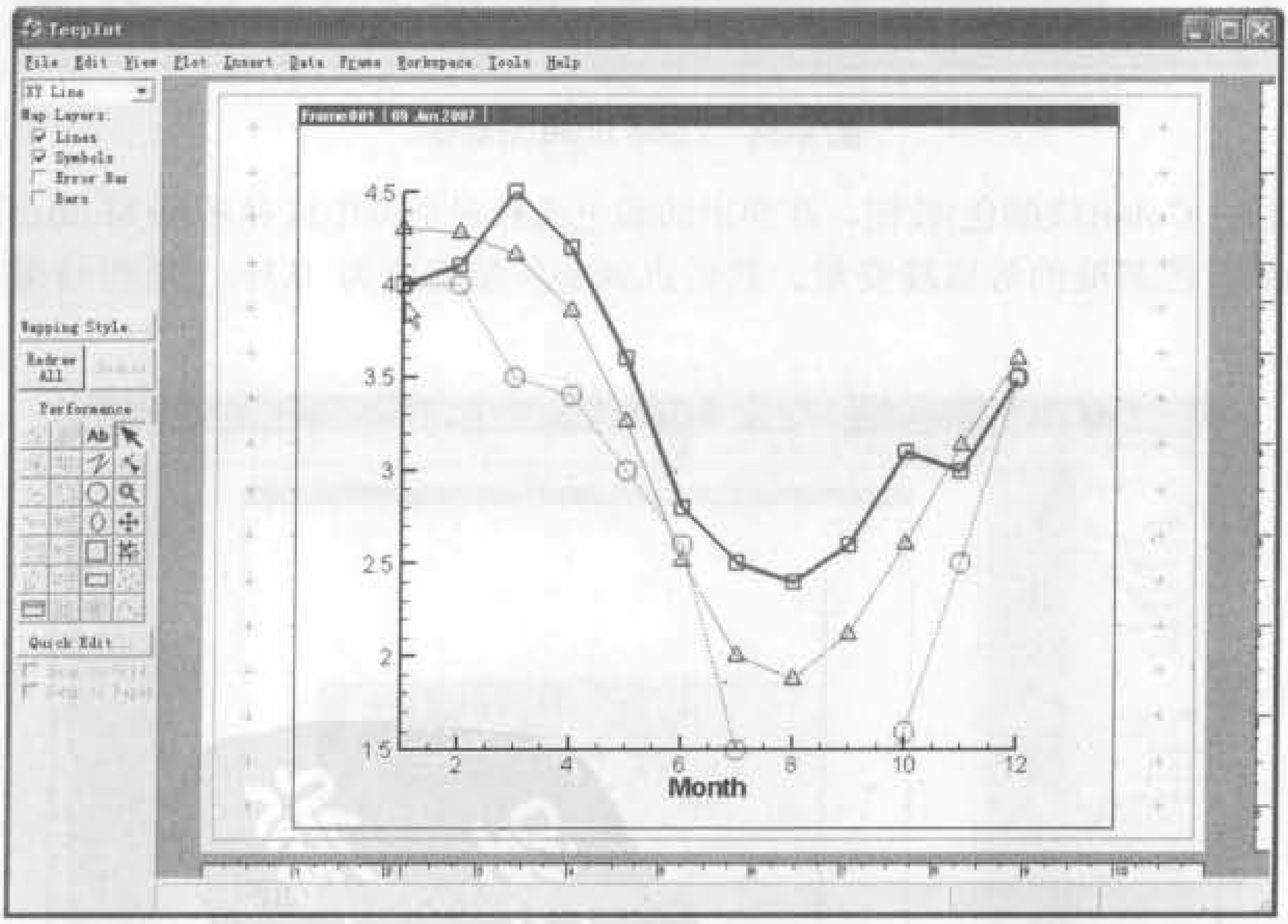


图 4-33　XY 曲线结果

4.5.2　绘制矢量图

(1) 选择 File→New Layout 命令。

(2) 选择 File→Load Datafile 命令，在打开的对话框中选择 Tecplot 的安装目录 TEC100 下的“Demo\2D\velocity.plt”。

(3) 在工具栏中，取消选中 Mesh 复选框，选中 Vector 复选框。在弹出的 Select Variables

对话框中设置 U 为 U/RFC，V 为 V/RFC，单击 OK 按钮。

(4) 由于默认的颜色为红色，不能反映出速度的大小，所以下面选择新的矢量着色。选择工具栏中的 Plot Attribute(绘图属性)，单击 Zone Style 按钮，打开如图 4-34 所示的 Zone Style 对话框。

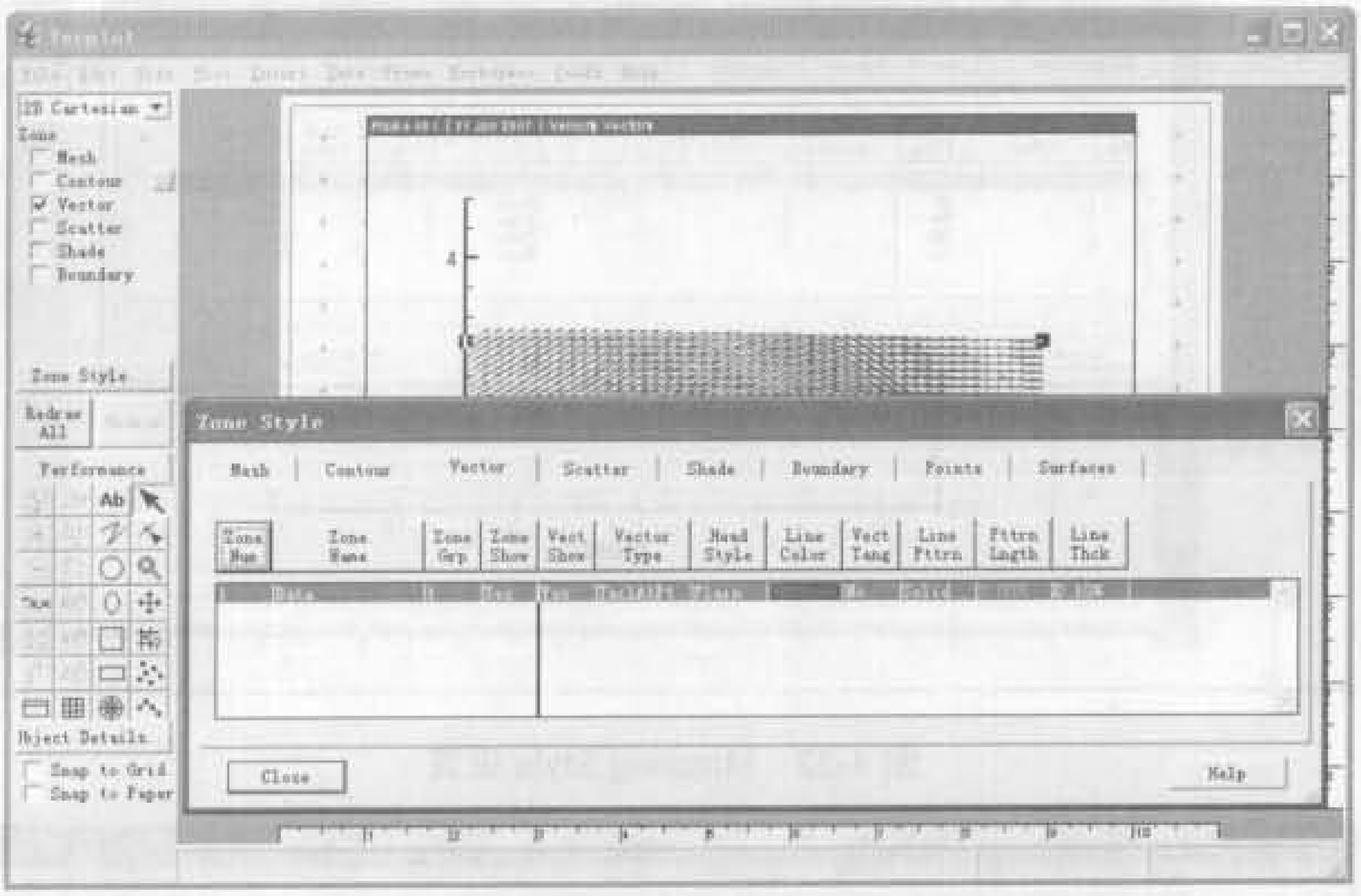


图 4-34 Zone Style 对话框

单击 Line Color(线颜色)按钮，在弹出的颜色选择对话框中选择多色(Multi)或者其他单色，并选择着色基准的等值线变量。然后再调整矢量宽度为 0.3%。所得结果如图 4-35 所示。

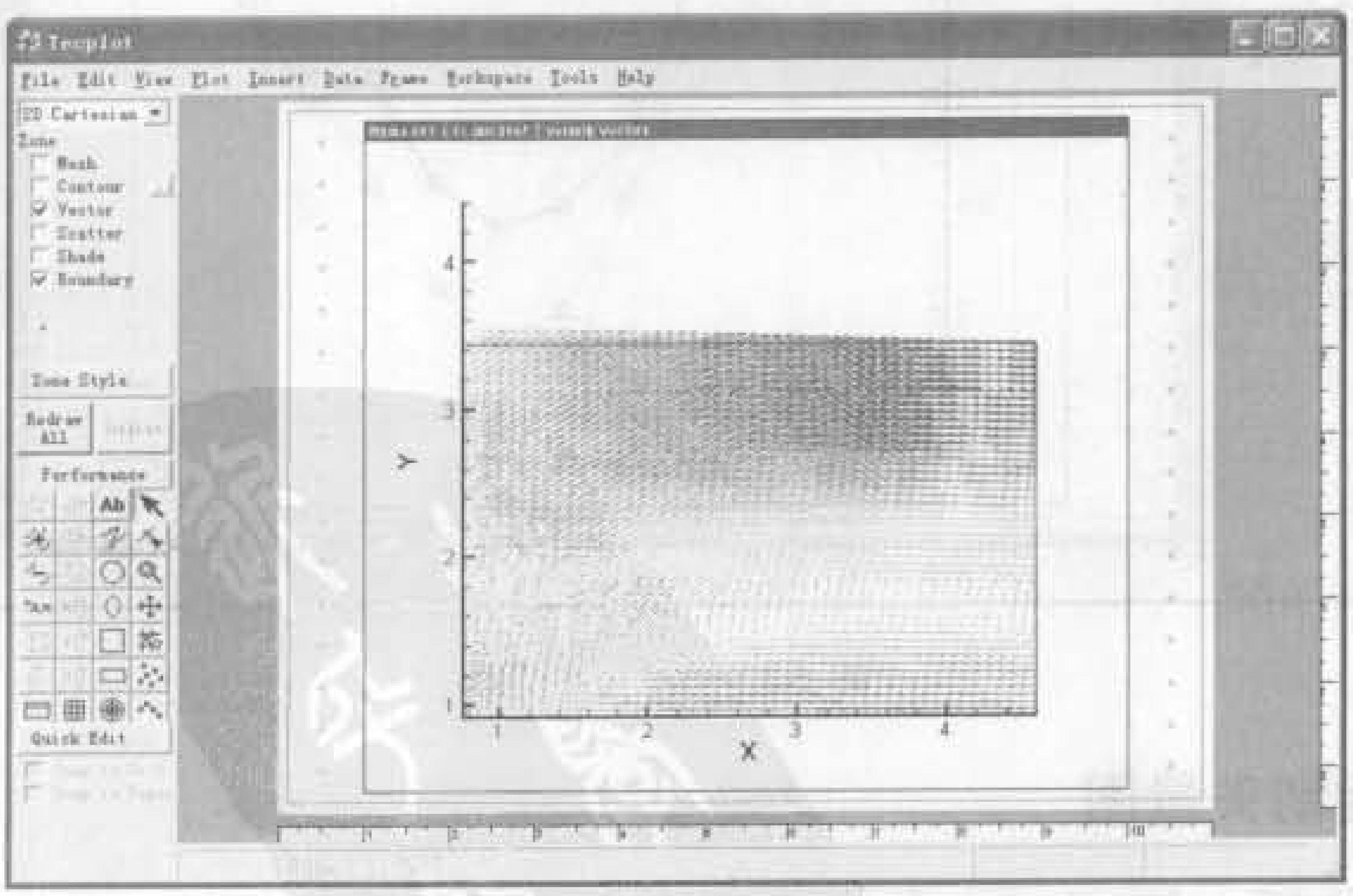


图 4-35 等值线颜色着色矢量图

(5) 还可以在矢量图上绘制流动迹线。在工具栏中单击按钮，然后在矢量图中上下拖动鼠标，便可以绘制出如图 4-36 所示的迹线图。

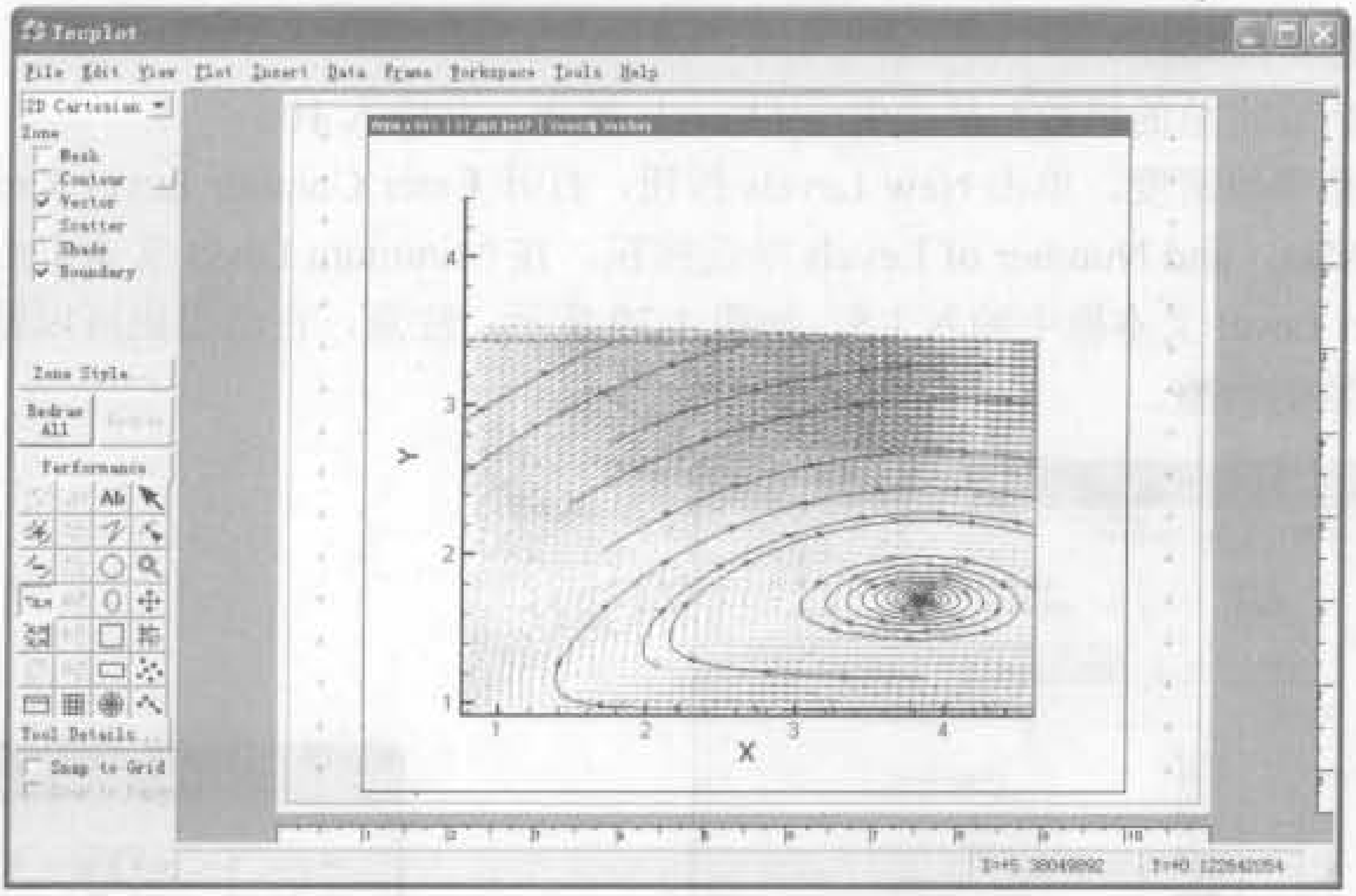


图 4-36　迹线图

4.5.3　绘制等值线图

为了保证示例的连贯性，这里依然应用 velocity.plt 文件。

(1) 选择 File→New Layout 命令。

(2) 选择 File→Load Data File 命令，在打开的对话框中选择 Tecplot 的安装目录 TEC100 下的“Demo\2D\velocity.plt”。

(3) 在工具栏中取消选中 Mesh 复选框，选中 Contour 复选框，在弹出的变量选择对话框中选择 R/RFR。注意，等值线变量也可以在 Field→Contour→Contour variables 中修改。生成的图形应如图 4-37 所示。

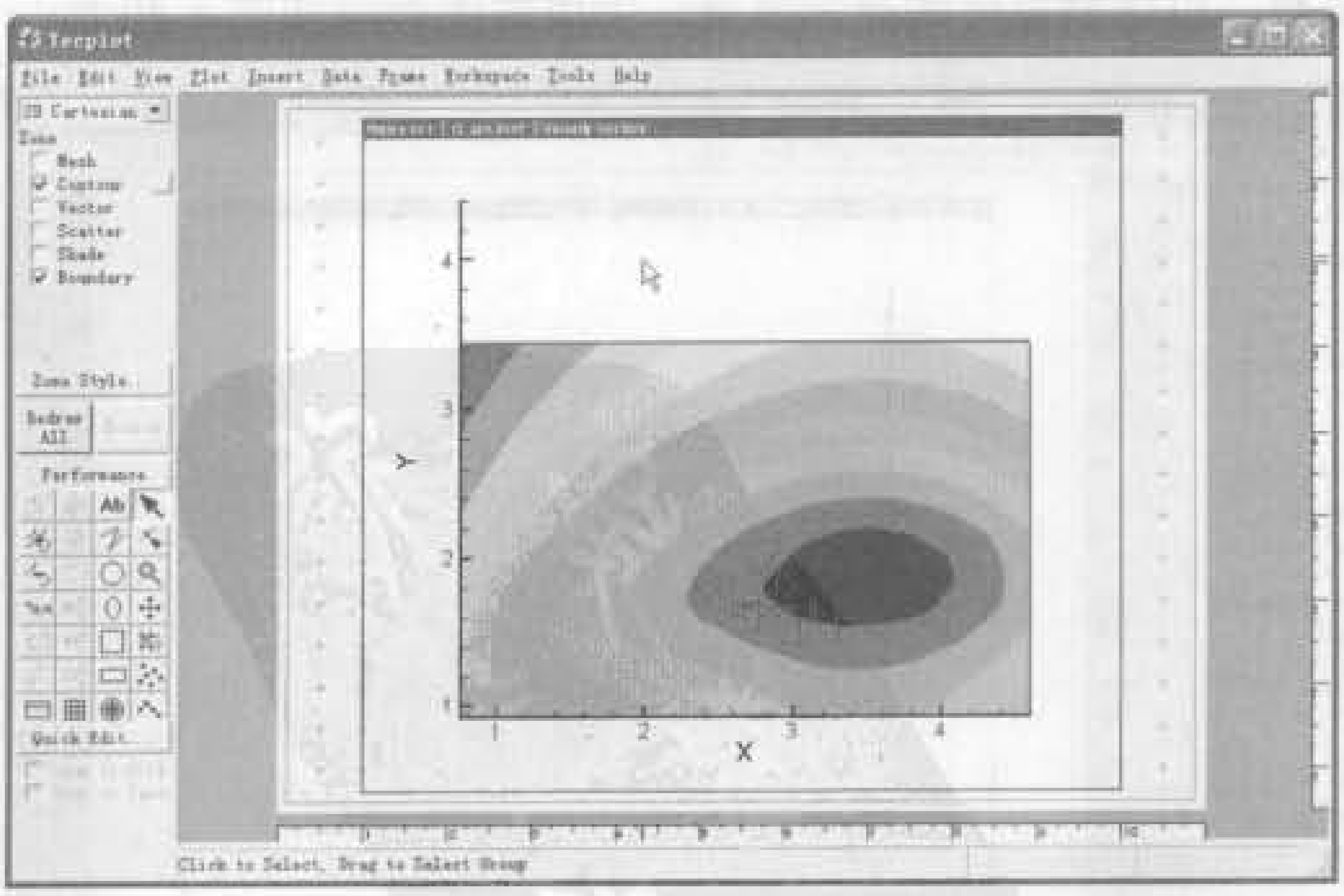


图 4-37　等值线图

(4) 下面调整等值线的密度和数值。

① 选择 Polt→Contour 命令，单击 More 按钮，会打开如图 4-38 所示的 Contour Details(等

值线详细信息)对话框。

② 重新设定总的级数。单击 Reset Levels 按钮，并输入 10。

③ 调整等级密度。单击 New Levels 按钮，打开 Enter Contour Level Range 对话框，选中 Min，Max，and Number of Levels 单选按钮，在 Minimum Level 文本框中输入 1.05，在 Maximum Level 文本框中输入 1.5，如图 4-39 所示。注意，在这里也可以输入级数来进行等值线级数的调整。

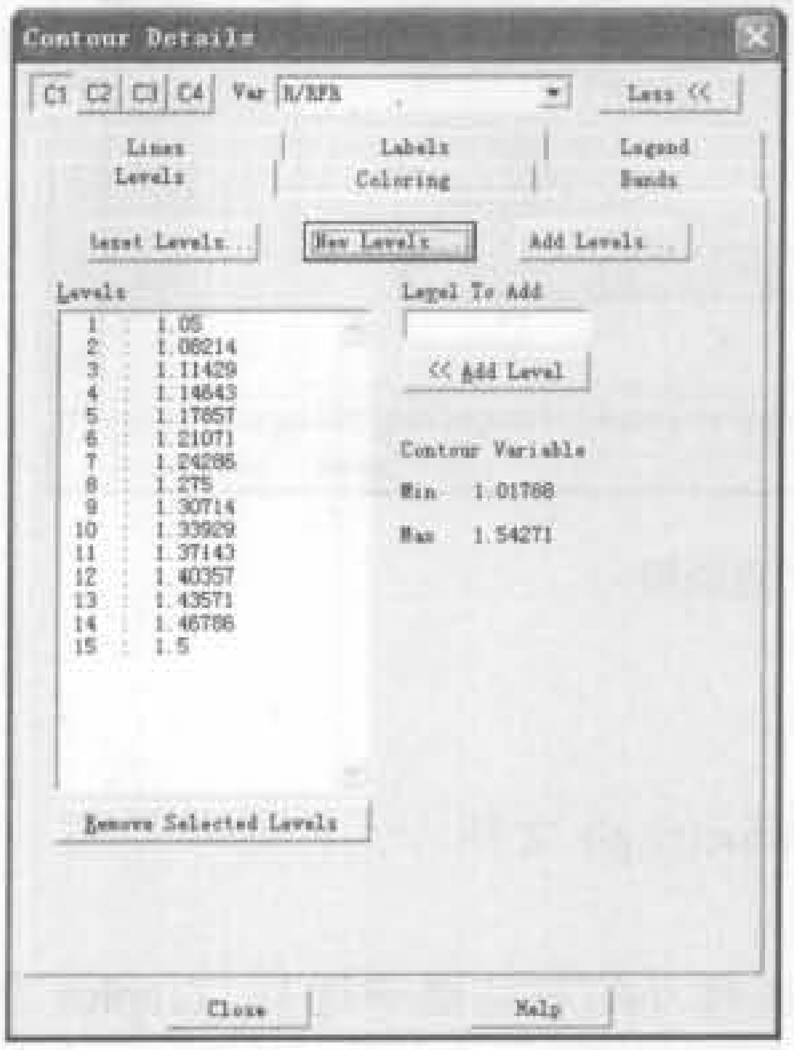


图 4-38　等值线等级调整的设置界面



图 4-39　调整等值线密度

(5) 加上等值线的轮廓线。单击工具栏中的 Zone Style 按钮，在弹出的对话框中单击 Contour Type 按钮，选择 Both Lines & Flood，等值线图形中便有了轮廓线，可以调整线型、颜色和宽度等参数，以达到最佳的显示效果。生成的图形应如图 4-40 所示。

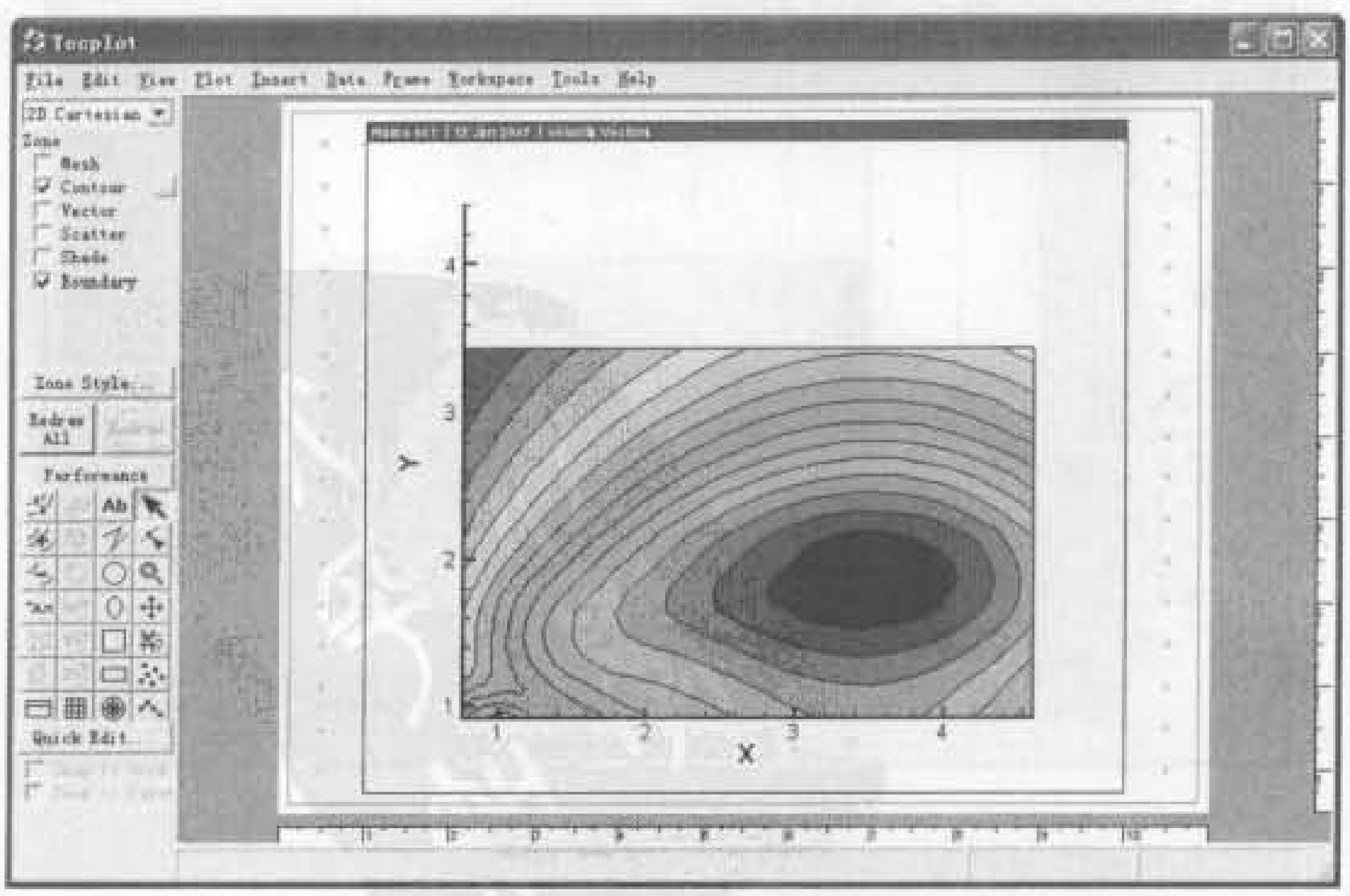


图 4-40　调整后的等值线及其轮廓线

(6) 加上图例并在轮廓线上标出数值。

① 选择 Plot→Contour 命令，在打开的对话框中单击 More 按钮，并切换到 Legend 选项卡，弹出如图 4-41 所示的界面。

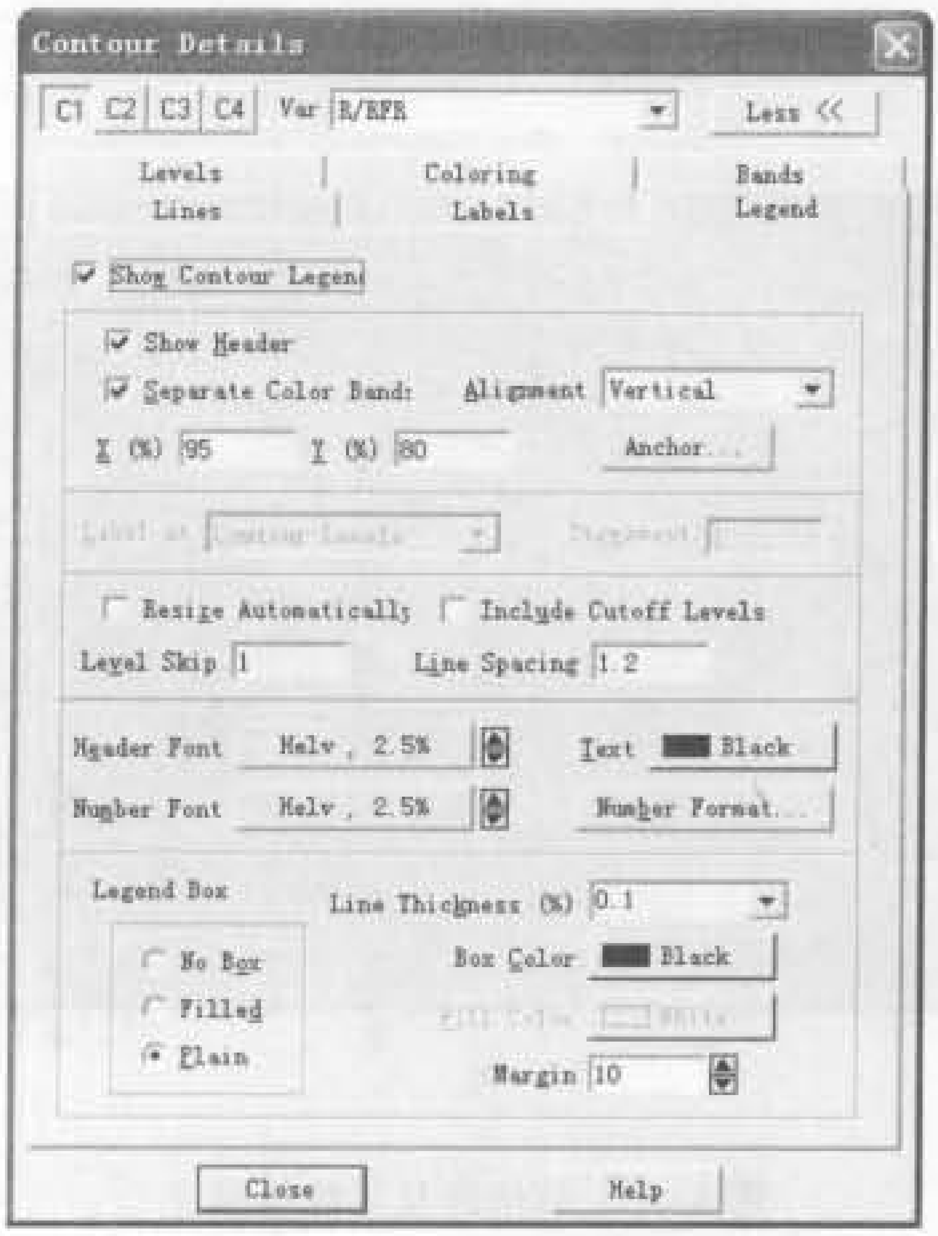


图 4-41　图例设置界面

② 选中 Show Contour Legend 复选框。

③ 选择 Align Vertical(垂直对齐)，可以通过 size(大小)调整其尺寸，以及通过 Box around table(是否有边框)确定是否有边框。调整后的窗口应如图 4-42 所示。

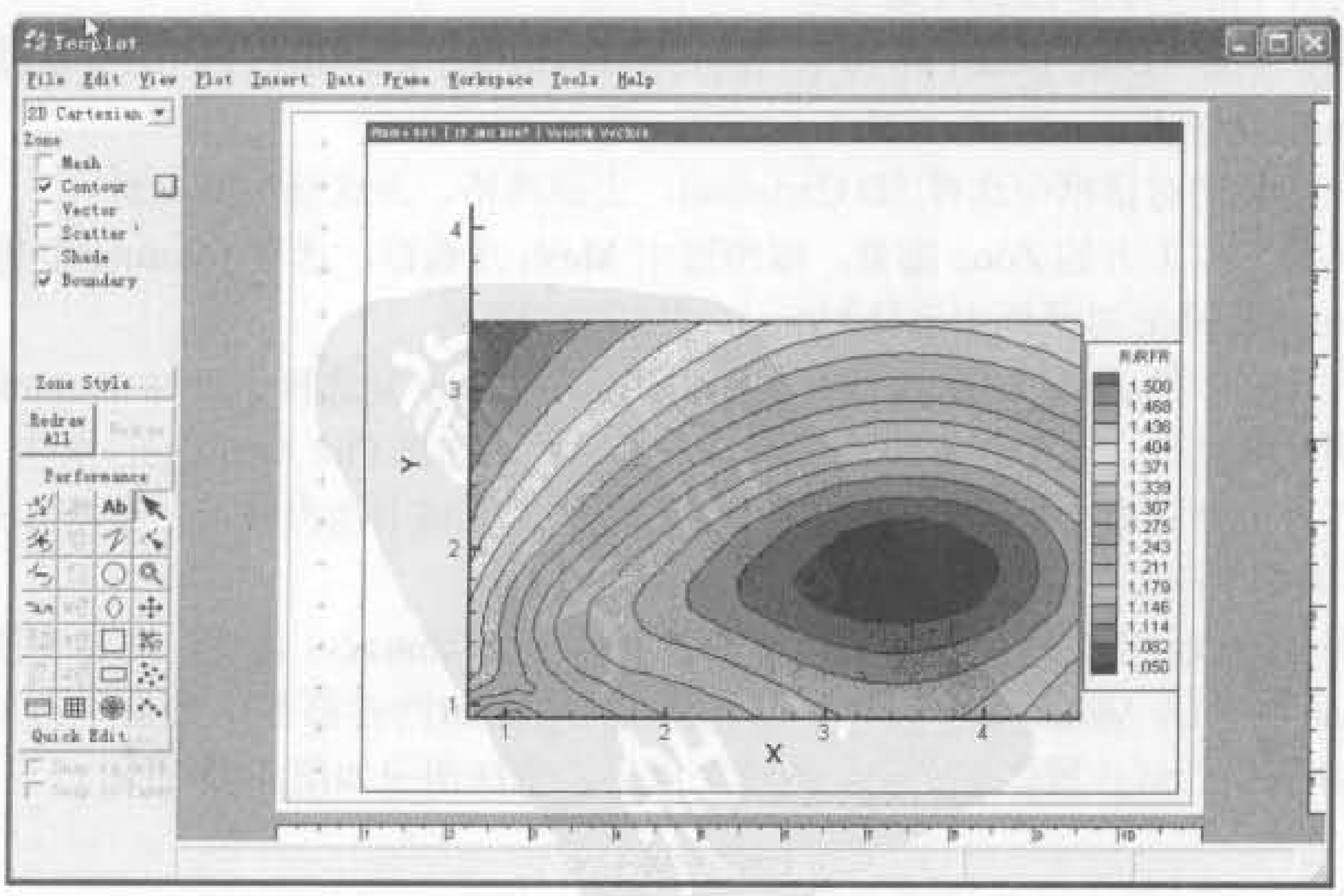


图 4-43　等值线及其图例

④ 单击工具栏中的 按钮，在希望标注的地方单击，便可以对等值线进行数字标注。全部操作完成后的图形应如图 4-43 所示。

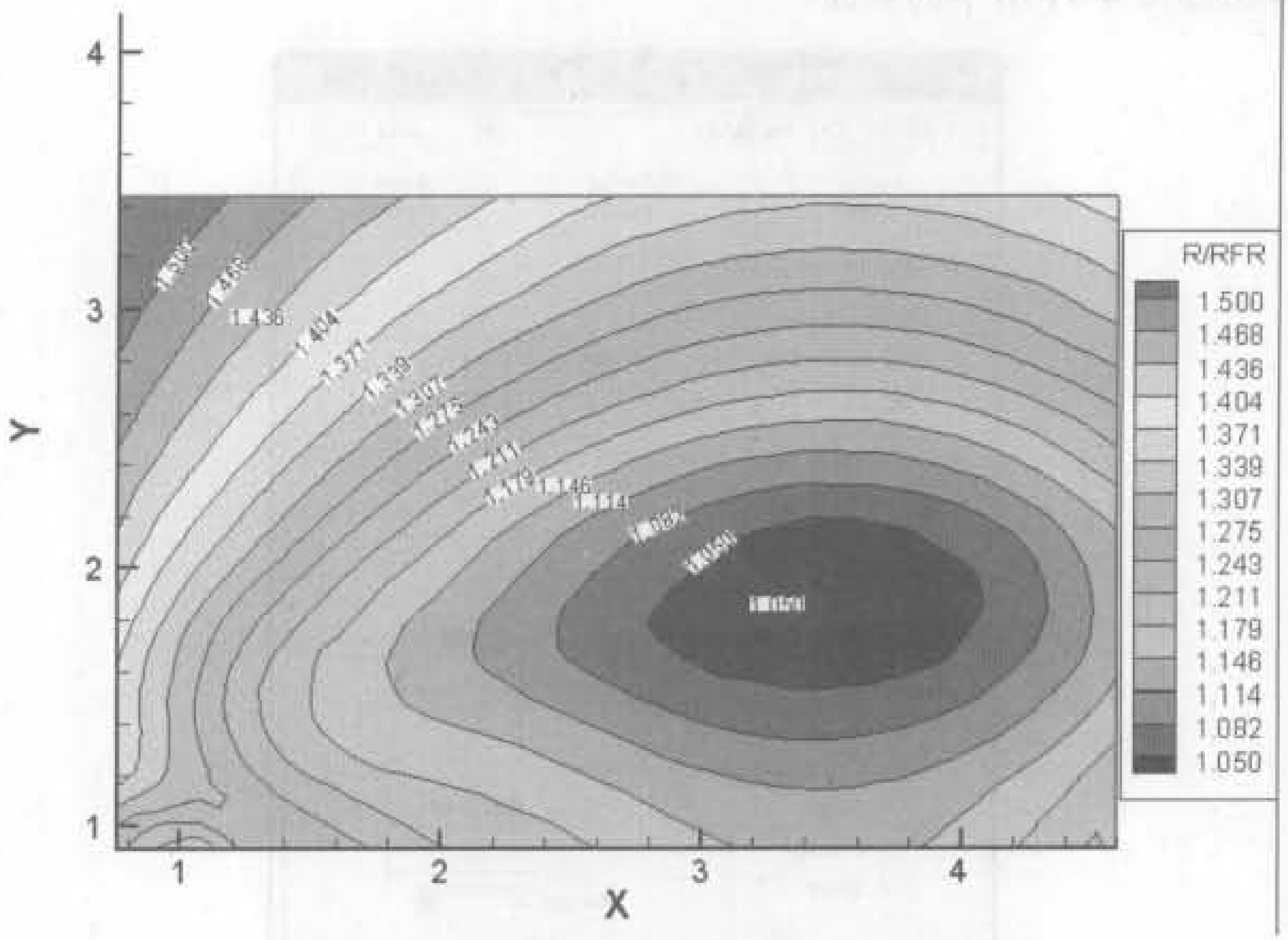


图 4-43 等值线及其数值标注

4.5.4 绘制流线图

在绘制矢量图的过程中，已经绘制了一个简单的2D流线图。本节将绘制更为复杂的3D体流线图。

(1) 选择 File→New Layout 命令。

(2) 选择 File→Load Data File 命令，在打开的对话框中选择 Tecplot 的安装目录 TEC90 下的"Demo\3D_Volume\ductflow.plt"。

(3) 在其后的对话框中选择 3D Cartesian，去掉网格，并选择矢量变量。

(4) 在边栏左上方的 Zone 部分，取消选中 Mesh 复选框，选中 Contour 复选框，在弹出的 Contour Variable 对话框中选择 U(ms)，关闭对话框。

(5) 在边栏中的 Zone Effects 部分，选中 Translucency 复选框，这样 Tecplot 会对整个绘图进行透明化处理，透明程度的设定可以在工具栏中选择 Plot Attribute，单击 Zone Style 按钮，在打开的对话框中的 Effects 选项中进行设定。完成书面操作后，整个 Tecplot 窗口应如 图 4-44 所示。

(6) 添加流线。首先设定流线属性。选择 Plot→Streamtraces 命令，在打开的对话框中设定 Format(格式)为 Volume Rod。Direction(方向)为 Both(两者都有)。然后单击工具栏中的 按钮，在图中合适位置添加流线。操作完成后的流线图应如图 4-45 所示。

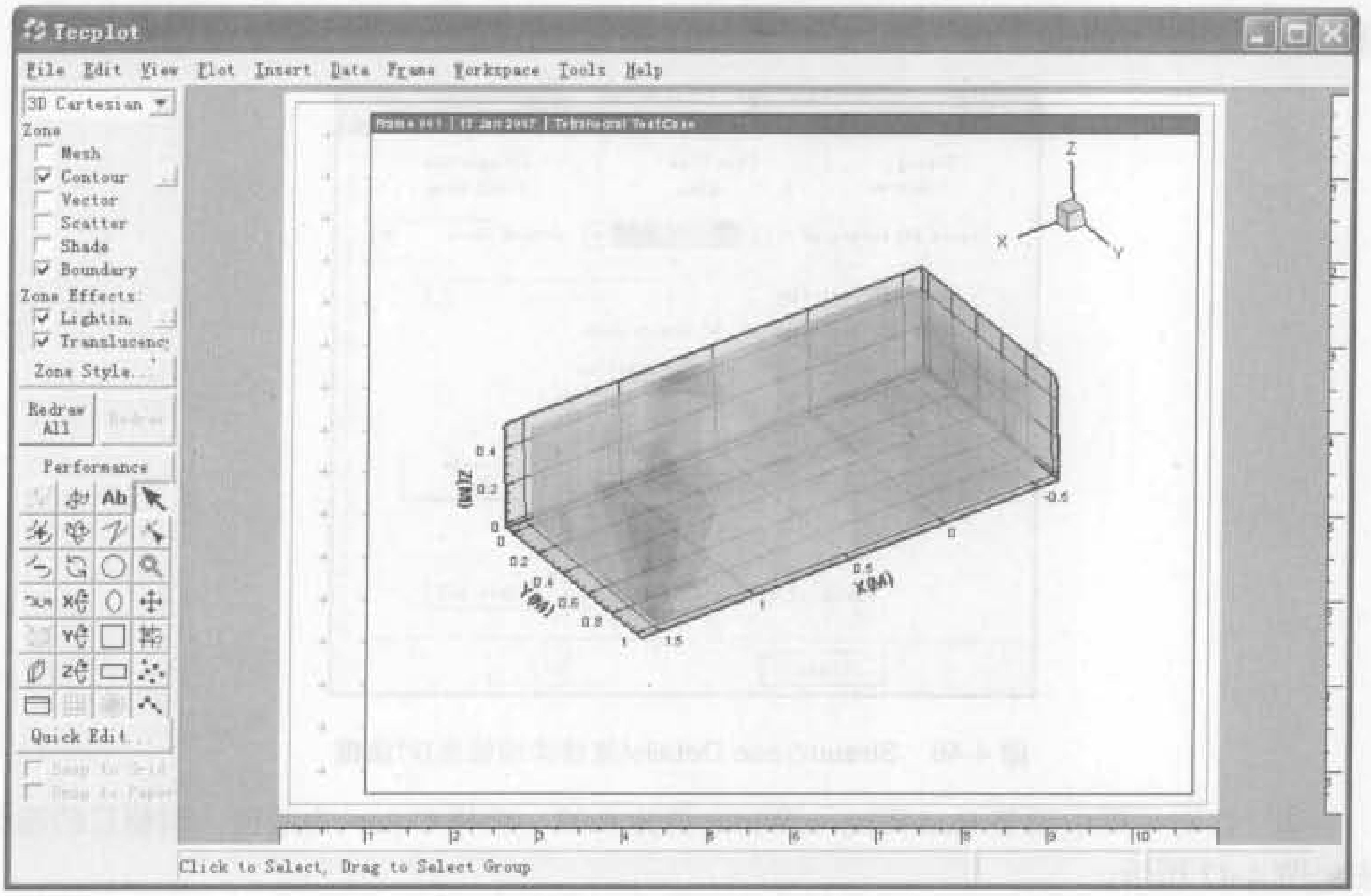


图 4-44　等值线图

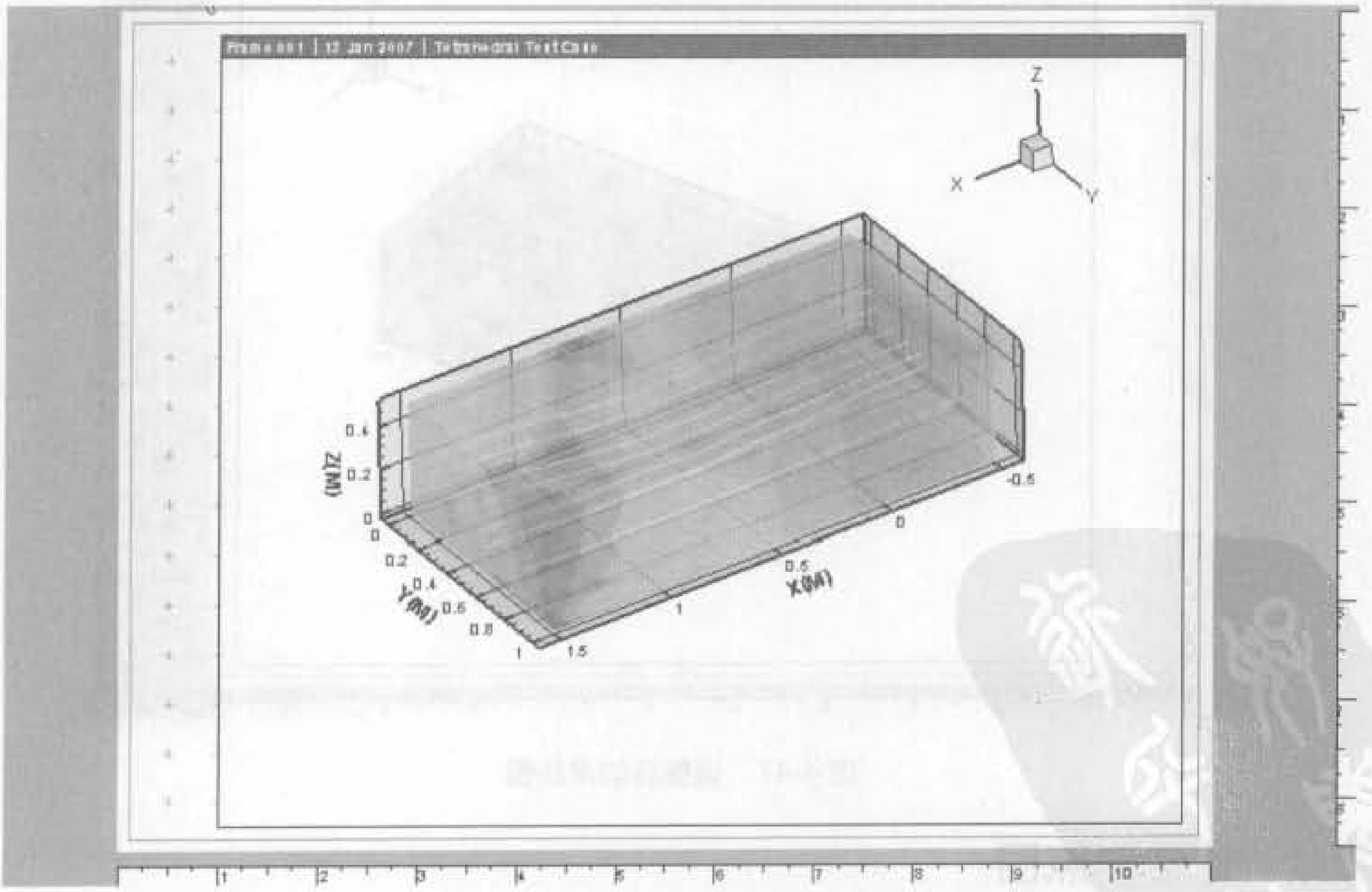


图 4-45　添加流线后的窗口

由于流线默认为白色，且宽度值太小，所以要作进一步调整。

① 选择 Plot→Streamtrace 命令，打开如图 4-46 所示的 Streamtrace Details(流线详细信

息)对话框。

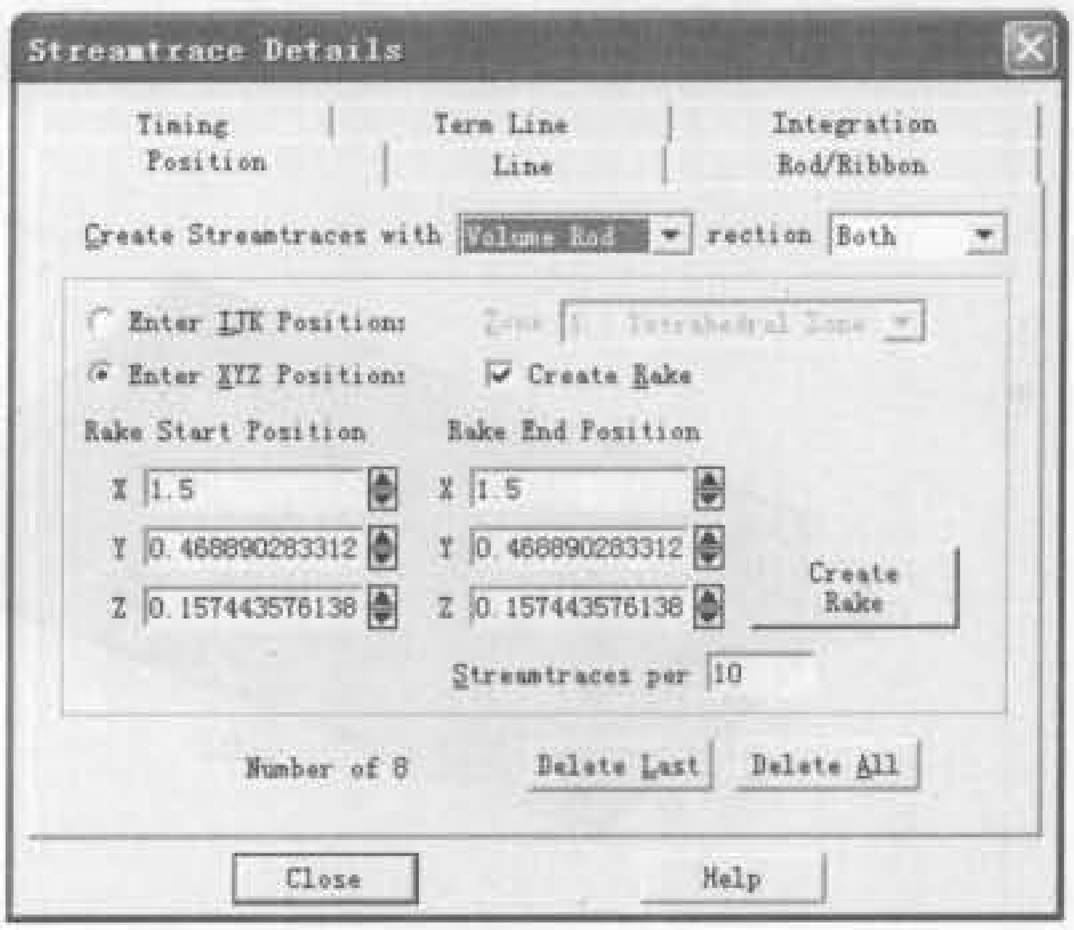


图 4-46 Streamtrace Details(流线详细信息)对话框

② 在对话框中调整 Rod/Ribbon Width 值为 0.03，选择 Color 为蓝色。调整后的流线图如图 4-47 所示。

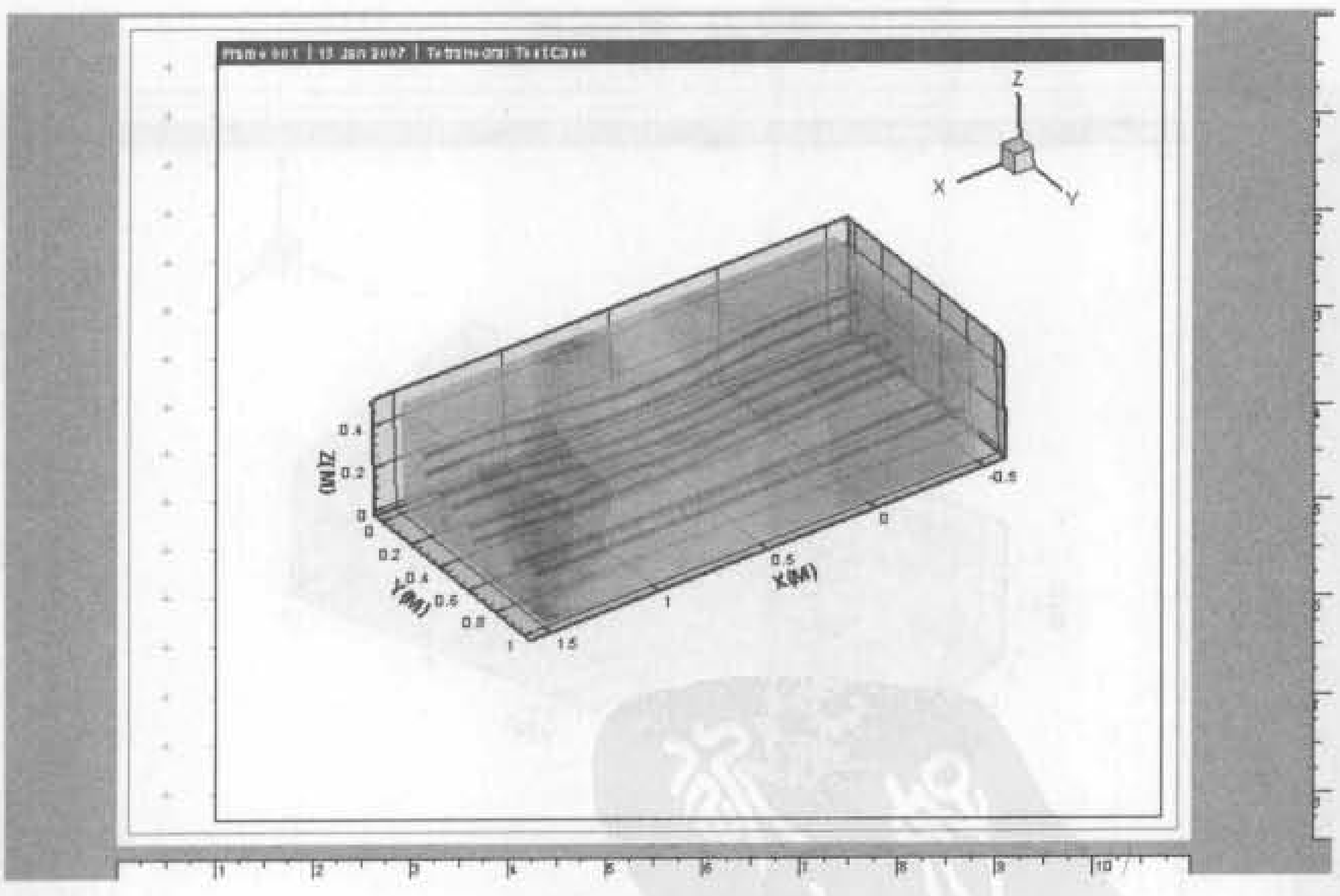


图 4-47 调整后的流线图

4.5.5 绘制散点图

所谓散点图是用一系列位于数值点位置的符号来表示数据的一种图形。这些代表数值的符号可以根据变量值的大小而有大小或者颜色的不同，当然也可以是相同大小或者相同颜色。与等值线图相比散点图最大的优势在于其不需要网格结构，可以对不规则的数据进

行处理。散点图的绘制相对简单，但要想得到很好的标识数据的散点图需要进行大量的调整工作。

(1) 选择 File→New Layout 命令。

(2) 选择 File→Load Data File 命令，在打开的对话框中选择 Tecplot 的安装目录 TEC90 下的“Demo\2D\cylinder.plt”。

(3) 取消选中 Mesh 复选框，并选中 Scatter 复选框。Tecplot 窗口应如图 4-48 所示(注：进行了位置调整，中心显示)。

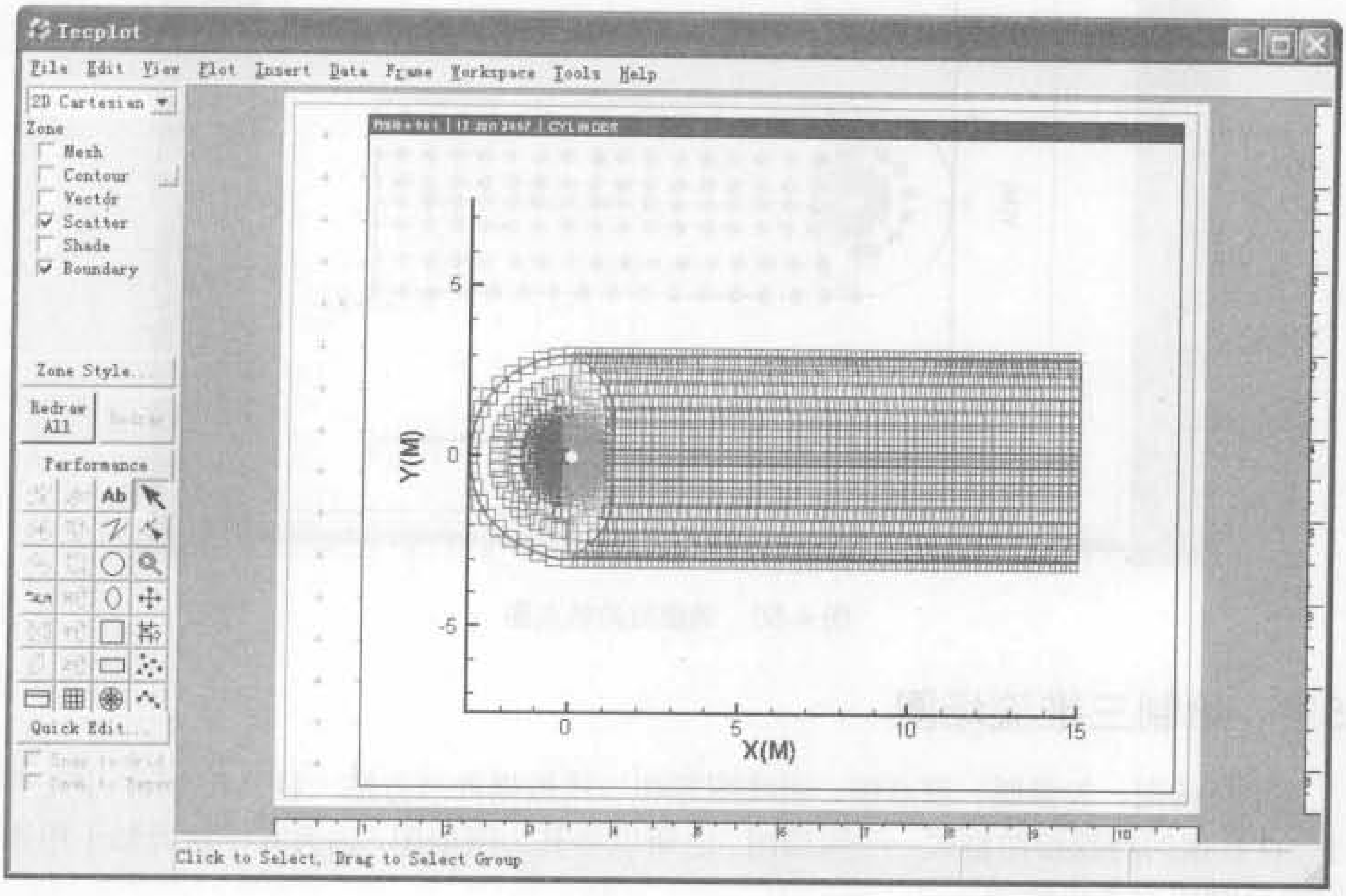


图 4-48 散点图

(4) 进行散点图调整。单击在工具栏中的 Zone Style 按钮，选择 Scatter 选项，出现对话框如图 4-49 所示。

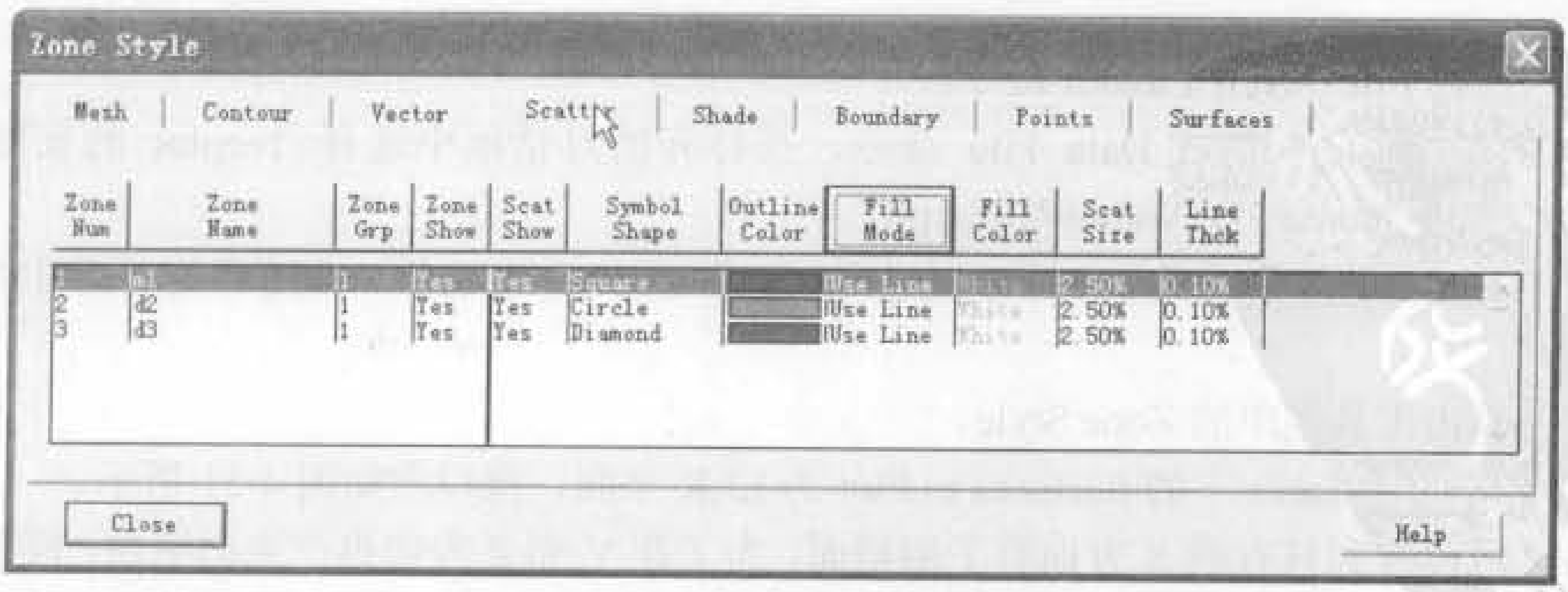


图 4-49 散点图调整的界面

(5) 由于点太密，根本无法观察，所以首先调整数据密度。单击 Point 对话框中的 Index

Step，调整 I、J 的 skip 值都为 3。

(6) 对三个区分别采用不同的符号。对 1 区采用默认方形(square)，对 2 区采用圆(circle)，对 3 区采用菱形(diamond)，并调整它们的轮廓颜色和填充色。调整后的散点图应如图 4-50 所示。

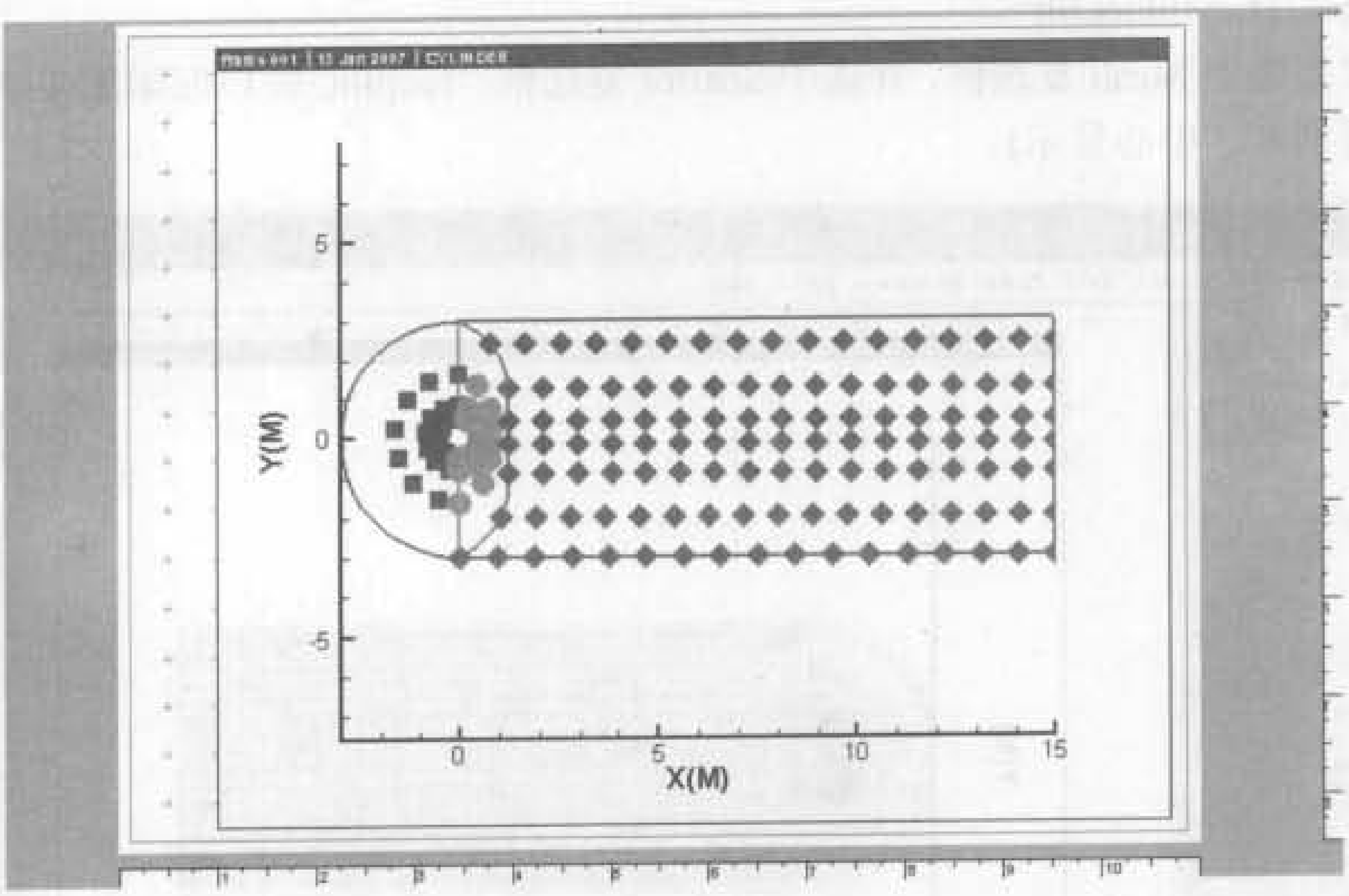


图 4-50　调整后的散点图

4.5.6　绘制三维流场图

三维流场图、矢量图、散点图、流线图等和二维数据处理方法一样。Tecplot 中还有针对于 3D 数据的特殊绘图格式——围墙图。之所以称其为围墙图，因为其形状类似于用围墙把平面分割成许多小格子。

围墙图的生成过程也非常简单，数据可以是 IJ 有序数据或者 I,J,K 平面和 IJK 数据。这种图形的特别之处在于要绘出底面数据，然后绘出垂直于底面的数据作为围墙，把面给分成小格子。可以用泛色的等值线图来充分表现数据的变化。下面结合数据来制作一个围墙图。

(1) 选择 File→New Layout 命令。

(2) 选择 File→Load Data File 命令，在打开的对话框中选择 Tecplot 的安装目录 TEC100 下的“Demo\3D_Volume\fluid.plt”。

(3) 在工具栏中取消选中 Mesh 复选框，选中 Contour 复选框。在随后的对话框中选择变量 E。

(4) 单击工具栏中的 Zone Style。

(5) 调整 Surfaces 下的 Surfaces to Plot 为 I,J,K 平面。窗口应如图 4-51 所示。

(6) 可以看到只有沿 X 方向有了围墙面，为了在 Y 和 Z 方向也产生围墙面，需要改变其范围为最大。在 Zone Style 对话框中的 Surfaces 选项卡下的 Range for I-Planes (I 平面范围)、Range for J-Planes(J 平面范围)和 Range for K-Planes(K 平面范围)中分别作如下调整。

I：Begin(起始)1；End(终止)Mx；Skip(跳过)5。

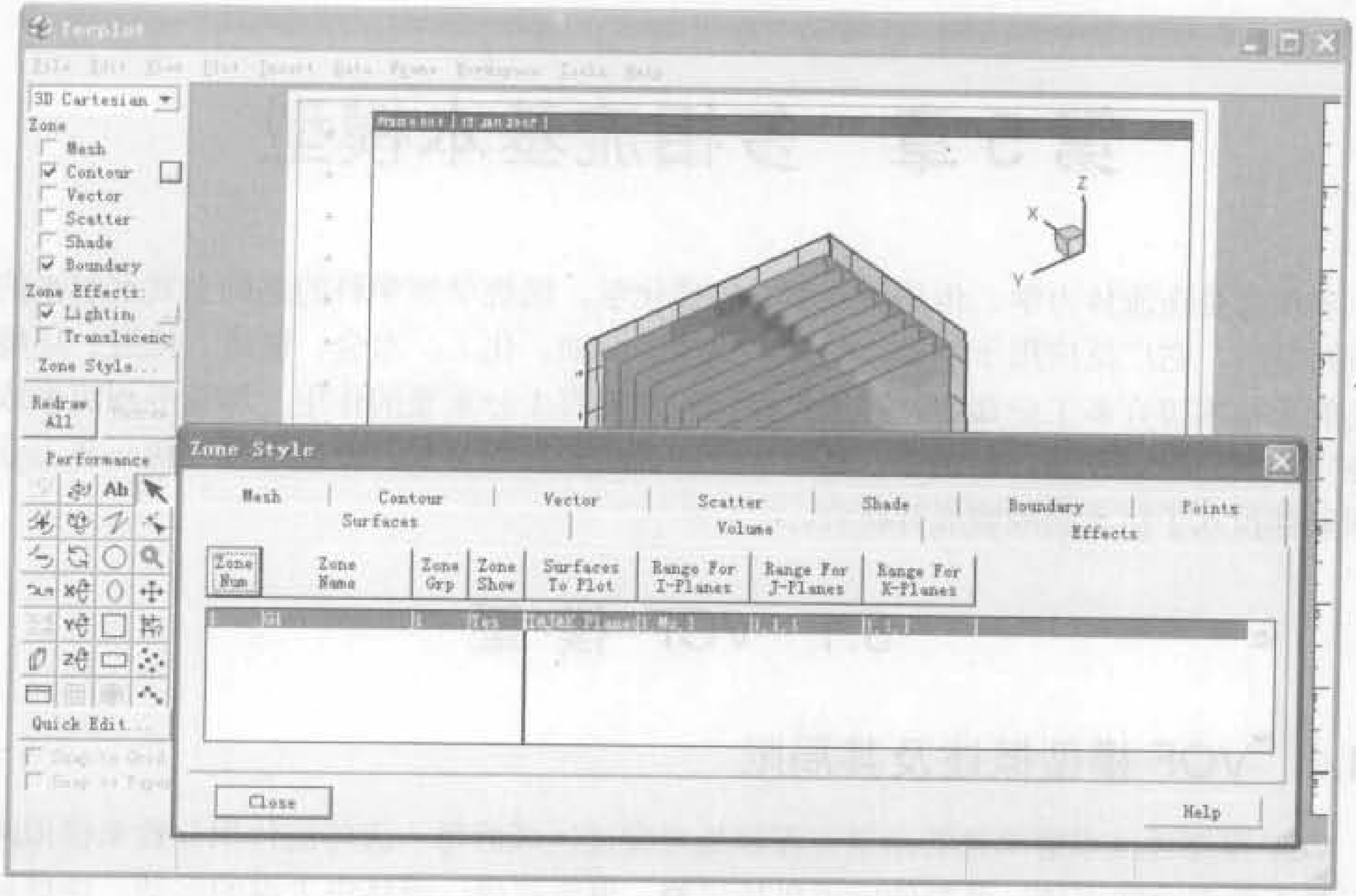


图 4-51　Sufaces 设置

J：Begin(起始)1；End(终止)Mx；Skip(跳过)4。

K：Begin(起始)1；End(终止)Mx；skip(跳过)Mx。

生成的图形如图 4-52 所示。

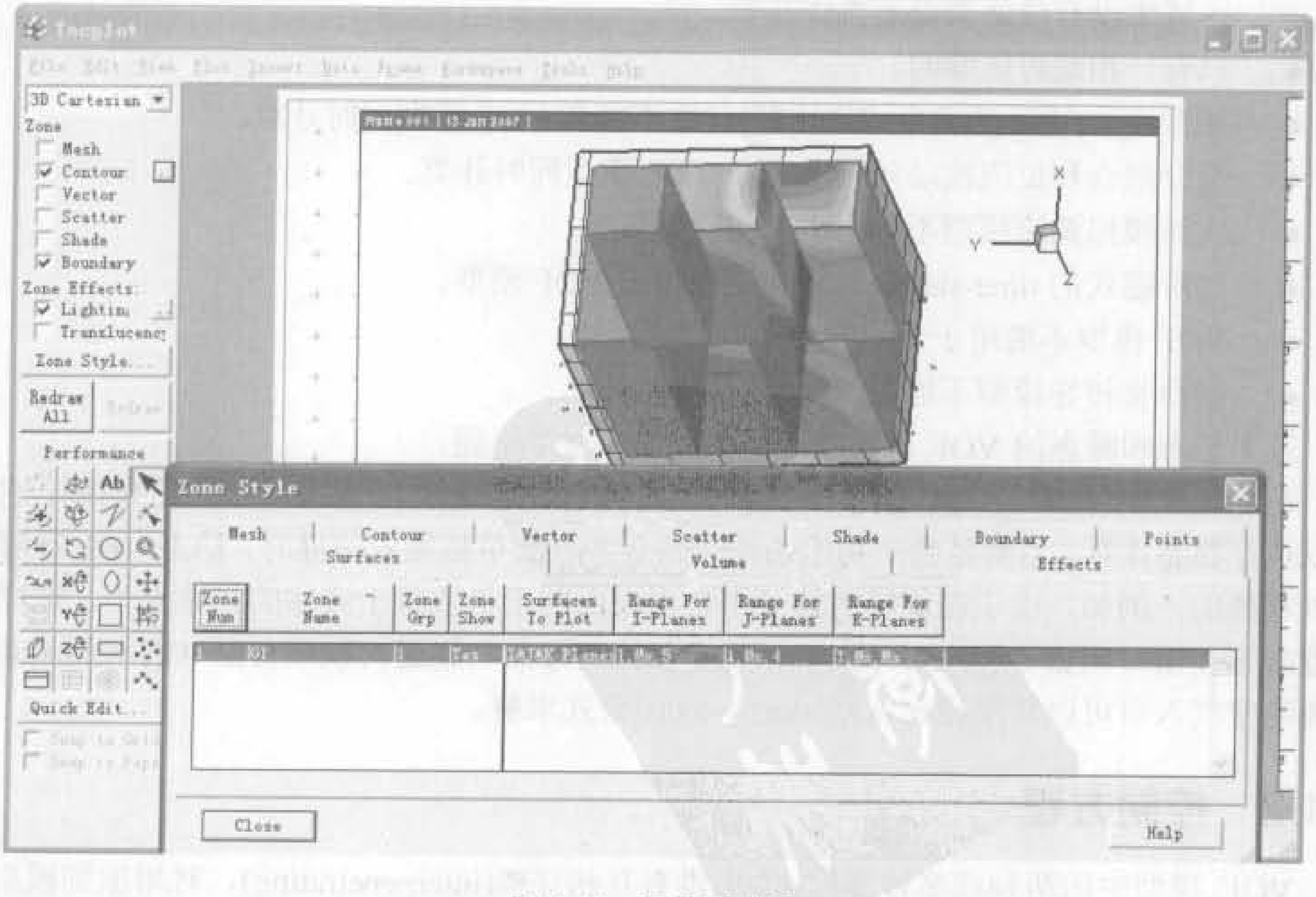


图 4-52　围墙图结果

第 5 章　多相流基本模型

多相流是在流体力学、传热传质学、物理化学、燃烧学等学科的基础上发展起来的一门新兴学科，它广泛应用于能源、动力、核能、石油、化工、冶金、制冷、低温、环境保护及航天技术等许多工业部门，对国民经济的发展有十分重要的作用。尽管多相流形成一门独立学科的时间不长，但由于它具有非常强的工业应用背景，发展速度很快，各工业发达国家都投入了巨大的经费展开研究。

5.1 VOF 模型

5.1.1 VOF 模型概述及其局限

VOF 模型通过求解单独的动量方程和处理穿过区域的每一流体的体积分数来模拟两种或三种不能混合的流体。典型的应用包括预测、射流破碎、流体中大泡的运动、决堤后水流动和气液界面的稳态和瞬态处理。

在 Fluent 中使用 VOF 模型存在以下一些限制。

- 必须使用离散求解器，VOF 模型不能用于耦合求解器。
- 所有的控制体积必须充满单一流体相或者相的联合，VOF 模型不允许在那些空的区域中没有任何类型的流体存在。
- 只有一相是可压缩的。
- 周期流动(比质量流率或比压降)问题不能和 VOF 模型同时计算。
- 组分混合和反应流动问题不能和 VOF 模型同时计算。
- 大涡模拟湍流模型不能用于 VOF 模型。
- 二阶隐式的 time-stepping 公式不能用于 VOF 模型。
- VOF 模型不能用于无粘流。
- 壁面壳传导模型不能和 VOF 模型同时计算。

关于稳态和瞬态的 VOF 计算，有如下需要注意的问题。

在 Fluent 中 VOF 公式通常用于计算时间依赖解，但是对于只关心稳态解的问题，它也可以执行稳态计算。当解是独立初始条件并且有明显的单相流入边界时，稳态 VDF 计算是可以选择的。例如，由于在旋转的杯子中自由表面的形状依赖于流体的初始水平，这样的问题必须使用时间依靠(time-dependent)公式。另一方面，渠道内顶部有空气的水的流动和分离的空气入口可以采用稳定状态(steady-state)公式求解。

5.1.2 控制方程

VOF 模型中的两种或多种流体(或相)没有互相穿插(interpenetrating)，对增加到模型里的每一附加相，就引进一个变量，即计算单元里的相的体积分数(the volume fraction of the

phase)。在每个控制体积内，所有相的体积分数的和为 1。所有变量及其属性的区域被各相共享并且代表了体积平均值(volume-averaged values)，只要每一相的体积分数在每一位置是可知的。这样，在任何给定单元内的变量及其属性可能是纯粹其中一相的变量及其属性，或者多相混合的变量及其属性，这取决于体积分数值。换句话说，在单元中，如果第 q 相流体的体积分数记为 α_q，那么就会出现下面三个可能的情况。

- $\alpha_q = 0$：第 q 相流体在单元中是空的。
- $\alpha_q = 1$：第 q 相流体在单元中是充满的。
- $0 < \alpha_q < 1$：单元中包含了第 q 相流体和一相或者其他多相流体的界面。

基于 α_q 的局部值、适当的属性和变量在一定范围内分配给每一控制体积。

1. 体积分数方程

在 VOF 模型中，跟踪相与相之间的界面是通过求解一相或多相的体积分数的连续方程来完成的。对第 q 相，这个方程如下：

$$\frac{\partial \alpha_q}{\partial t} + v_q \cdot \nabla \alpha_q = \frac{S_{\alpha_q}}{\rho_q} + \frac{1}{\rho_q}\sum_{p=1}^{n}(\dot{m}_{pq} - \dot{m}_{qp}) \tag{5-1}$$

式中：$\dot{m}_{pq}$ 是 p 相到 q 相的质量输送，$\dot{m}_{qp}$ 是 q 相到 p 相的质量输送。默认情形，方程(5-1)右端的源项为零，但除了给定每一相指定常数或用户定义的质量源。体积分数方程(the volume fraction equation)不是为主相求解的，主相体积分数的计算基于如下的约束：

$$\sum_{q=1}^{n} \alpha_q = 1 \tag{5-2}$$

2. 属性

出现在输运方程中的属性(Properties)是由存在于每一控制体积中的分相决定的。例如，在两相流系统中，如果相用下标 1 和 2 表示，如果第二相的体积分数被跟踪，那么每一单元中的密度由式(5-3)给出：

$$\rho = \alpha_2 \rho_2 + (1 - \alpha_2)\rho_1 \tag{5-3}$$

通常，对 n 相系统，体积分数平均密度采用如下形式：

$$\rho = \sum \alpha_q \rho_q \tag{5-4}$$

所有的其他属性(例如粘度(viscosity))都以这种方式计算。

3. 动量方程

通过求解整个区域内的单一的动量方程(the momentum equation)，作为结果的速度场是由各相共享的。如式(5-5)所示，动量方程取决于通过属性 ρ 和 μ 的所有相的体积分数。

$$\frac{\partial}{\partial t}(\rho \vec{v}) + \nabla \cdot (\rho \vec{v}\vec{v}) = -\nabla p + \nabla \cdot [\mu(\nabla \vec{v} + \nabla \vec{v}^T)] + \rho \vec{g} + \vec{F} \tag{5-5}$$

共享区域近似的一个局限是，在各相之间存在大的速度差异的情形下，靠近界面的速度的计算精确性会受到不利影响。

4. 能量方程

能量方程(the energy equation)也是各相共享的，表示如下：

$$\frac{\partial}{\partial t}(\rho E)+\nabla\cdot[\vec{v}(\rho E+p)]=\nabla\cdot(k_{\mathrm{eff}}\nabla T)+S_h \tag{5-6}$$

VOF 模型处理能量 E 和温度 T，作为质量平均变量：

$$E=\frac{\sum_{q=1}^{n}\alpha_q\rho_q E_q}{\sum_{q=1}^{n}\alpha_q\rho_q} \tag{5-7}$$

这里对每一相的 E_q 是基于该相的比热和共享温度。

属性 ρ 和 k_{eff} (有效热传导)是被各相共享的。源项 S_h 包含辐射的贡献，也有其他体积热源。和速度场一样，在相间存在大的温度差时，靠近界面的温度的精确度也受到限制。在属性有几个数量级的变化时，这样的问题还会增长。例如，如果一个模型包括液态金属和空气，材料的导热性有四个数量级的差异。如此大的差异使得方程需要设置各向异性的系数，以达到收敛性和精度范围。

5. 附加的标量方程

依赖于需要求解问题的定义，在求解时或许涉及附加的标量方程(additional scalar equations)。在湍流情形时，只求解一套输送方程，湍流变量(例如 k-ε 或雷诺应力)被通过整个区域的各相所共享。

6. 界面附近的插值

Fluent 中的控制体积公式要求计算穿过控制体积面的对流和扩散通量并与控制体积本身内部的源项平衡。对于 VOF 模型，Fluent 中有 4 种方案计算面的通量：几何重建(geometric reconstruction)、物质接受(donor-acceptor)、欧拉显式和隐式。

在几何重建和物质接受方案中，Fluent 用了特殊的插值处理两相之间界面附近的单元。图 5-1 显示了用这两种方法计算过程中沿着假定的界面的实际界面的形状。

欧拉显式和隐式方案以相同的插值方式处理这些完全充满一相或其他相的单元(也就是使用标准迎风、二阶或者 QUICK 方案)，而不采用特殊的处理。

1) 几何重建方案

在几何重建方案(the geometric reconstruction scheme)中， Fluent 中使用的标准插值方案用于获得界面通量，无论何时单元被充满一相或其他相。当单元靠近两相之间的界面时，使用几何重建方案。

几何重建方案使用分段线性的方法描绘了流体之间的界面。Fluent 中这个方案是最精确的并适合于通用的非结构化网格。几何重建方案假定两流体之间的界面在每个单元内有个线性斜面，并使用这个线性形状来计算穿过单元面的流体的对流。

这个重建方案的第一步是根据体积分数和由单元引出的信息计算相对于每个部分充满单元的中心的线性界面的位置；第二步是使用已计算的关于面上的法向和切向速度分布的信息线性的界面描述来计算穿过每个面的流体的对流量；第三步是使用前面的步骤中计算

的通量平衡计算每个单元的体积分数。

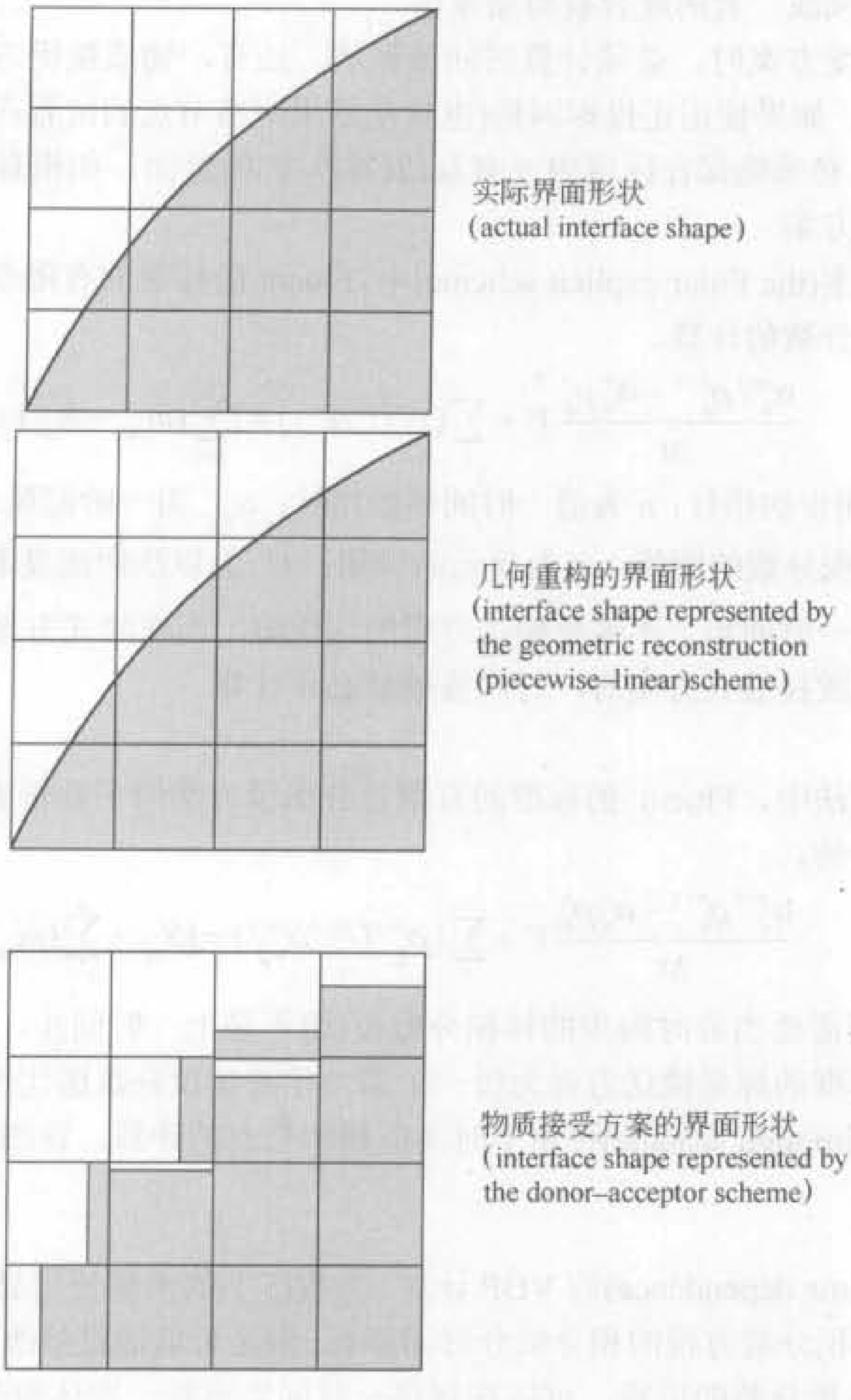


图 5-1　界面插值

当使用几何重建方案时，时间依赖解必须计算。同样，如果使用正投影网格(也就是如果网格节点的位置在两个子区域相交的边界上是一样的)，必须确保在区域内没有双边(零厚度)的壁面。如果有，必须分开它们。

2)　物质接受方案

在物质接受方案(the donor-acceptor scheme)中，Fluent 中使用的标准插值方案用于获得面的通量，无论何时单元内完全充满一相或其他相。当单元靠近两相之间的界面时，物质接受方案用于决定穿过面的流体的水平对流量。这个方案把一个单元看作一定数量的流体来自一相的捐赠(donor)，把另外(相邻)的单元看作相同数量流体的接受(acceptor)，这样使用防止了界面上的数值扩散。来自对流跨过一个单元边界一相流体的数量受限于两个值的最小值：捐赠单元的充满体积和接受单元的自由体积。

界面的方向也用于决定面的通量。界面的方向是水平的还是垂直的，取决于单元内第 q

相的体积分数梯度的方向和问题中共享面的相邻单元。依靠界面的方向和它的运动，通过纯的迎风、纯的顺风或二者的联合获得通量值。

当使用物质接受方案时，必须计算时间依赖解。还有，物质接受方案仅用于四边形和六面体网格。另外，如果使用正投影网格(也就是如果网格节点的位置在两个子区域相交的边界上是一样的)，必须确保在区域内没有双边(零厚度)的壁面。如果有，必须分开它们。

3) 欧拉显式方案

在欧拉显式方案(the Euler explicit scheme)中，Fluent 的标准的有限差分插值方案被用于前一时间步的体积分数的计算。

$$\frac{\alpha_q^{n+1}\rho_q^{n+1}-\alpha_q^n\rho_q^n}{\Delta t}V+\sum_f(\rho_q^n U_f^n \alpha_{q,f}^n)=[\sum_{p=1}^{n}(\dot{m}_{pq}-\dot{m}_{qp})+S_{\alpha_q}]V \tag{5-8}$$

式中：$n+1$ 为新时间步的指针；n 为前一时间步的指针；$\alpha_{q,f}$ 为一阶迎风、二阶迎风、QUICK 算法中的第 q 相体积分数的面值；V 为单元的体积；U_f 为以法向速度通过面的体积流量。

这个公式在每一时间步上不需要输送方程的迭代解，而在隐式方案中是需要的。

同样，当使用欧拉显式方案时，时间依赖解必须计算。

4) 隐式方案

在隐式插值方法中，Fluent 的标准的有限差分插值方案用于获得所有单元的面通量，包括那些界面附近的。

$$\frac{\alpha_q^{n+1}\rho_q^{n+1}-\alpha_q^n\rho_q^n}{\Delta t}V+\sum_f(\rho_q^{n+1} U_f^{n+1} \alpha_{q,f}^{n+1})=[S_{\alpha_q}+\sum_{p=1}^{n}(\dot{m}_{pq}-\dot{m}_{qp})]V \tag{5-9}$$

由于这个方程需要当前时间步的体积分数值(而不是上一时间步，如欧拉显式方案)，在每一时间步内标准的标量输送方程为每一个第二相的体积分数迭代性地求解。

隐式方案(the implicit scheme)可用于时间依赖和稳态的计算。详细内容见 5.6.3 节。

7. 时间依赖

对时间依赖(time dependence)的 VOF 计算，方程(5-1)的求解使用显式的时间匹配方案。Fluent 自动地为体积分数方程的积分细分时间步长，但是可以通过修改 Courant 数影响这个时间步长。对于体积分数的更新，可以选择每一时间步更新一次体积分数，或者每一时间步内的每一次迭代更新一次。

关于时间步选择详见 5.6.11 节。

8. 表面张力和壁面粘附

VOF 模型也可以模拟沿着每一对相之间的表面张力的影响，主要是通过附加的描述相和壁面之间的接触角来实现。

1) 表面张力

由于流体中分子之间的引力，产生了表面张力(surface tension)。例如，考虑水中的一个气泡，在气泡内，由于其周围相邻分子的作用，作用在分子上的净力为零。然而，在表面上，净力是放射状地向内的，跨过整个球面的径向分力的联合影响是表面收缩，因而增强了表面凹侧的压力。表面张力是一种仅作用在表面上的力，在这个例子中它必须是保持平衡的。它扮演了平衡内部放射状的分子引力和跨过表面的放射状的外部压力梯度的角色。

在两种流体分离的地区(但是它们之一不是球泡的形式)，表面张力的作用是通过减小界面的面积使自由能最小化。

Fluent 中表面张力模型是由 Brackbill 等提出的连续表面力模型。用这个模型，VOF 计算中附加的表面张力导致了动量方程中的源项。为了理解这个源项的起源，考虑沿着表面张力为常数的特殊情况，那些地方只考虑垂直于界面的力。可以看出，跨过表面的压降依赖于表面张力系数 σ 和通过两个半径的正交方向量度的表面曲率 R_1 和 R_2：

$$p_2 - p_1 = \sigma\left(\frac{1}{R_1} + \frac{1}{R_2}\right) \tag{5-10}$$

式中：p_1 和 p_2 是两种流体界面两侧的压力。

在 Fluent 中，使用 CSF 模型公式时，这里的表面曲率是从垂直于界面的表面的局部梯度计算的。n 为表面法线，定义 α_q 为第 q 相体积分数的梯度。

$$n = \nabla \alpha_q \tag{5-11}$$

表面曲率 κ 是为了区别单位法向量 $\hat{n}$ 而定义的：

$$\kappa = \nabla \cdot \hat{n} \tag{5-12}$$

式中：

$$\hat{n} = \frac{n}{|n|} \tag{5-13}$$

表面张力也可以根据越过表面的压力跃变给出。表面张力使用散度定理可以表示为体积力，正是这个体积力成了添加给动量方程的源项，它有如下形式：

$$F_{\text{vol}} = \sum_{\text{pairs } ij,\, i<j} \sigma_{ij} \frac{\alpha_i \rho_i \kappa_j \nabla \alpha_j + \alpha_j \rho_j \kappa_i \nabla \alpha_i}{\frac{1}{2}(\rho_i + \rho_j)} \tag{5-14}$$

该式允许在多于两相存在的单元附近力光滑地叠加。如果一个单元中只有两相，那么 $\kappa_i = -\kappa_j$，$\nabla\alpha_i = -\nabla\alpha_j$，方程(5-14)简化为

$$F_{\text{vol}} = \sigma_{ij} \frac{\rho \kappa_i \nabla \alpha_i}{\frac{1}{2}(\rho_i + \rho_j)} \tag{5-15}$$

式中：ρ 是使用方程(5-14)计算的体积平均密度。方程(5-15)显示了一个单元表面张力源项是与单元的平均密度成比例的。

注意，三角形和四面体网格上表面张力影响的计算不如四边形和六面体网格的计算精确，所以表面张力影响最重要的地区应当采用四边形和六面体网格。

2)　当表面张力的影响重要时

表面张力影响重要性的决定是基于两个无量纲数：雷诺数 Re 和毛细数(capillary number) Ca 或雷诺数 Re 和韦伯数(Weber number) We。当 $Re \ll 1$ 时，感兴趣的量是毛细数：

$$Ca = \frac{\mu U}{\sigma} \tag{5-16}$$

当 $Re \gg 1$ 时，感兴趣的是韦伯数：

$$We = \frac{\sigma}{\rho L U^2} \tag{5-17}$$

式中：U 是自由流速度。如果 $Ca \gg 1$ 或 $We \ll 1$，表面张力效应可以忽略。

3) 壁面粘附

与表面张力模型联合时选择指定一个壁面粘附(wall adhesion)角在 VOF 模型中也是有用的。这个模型假定流体与壁面产生的接触角常用于调整壁面附近单元表面的法向，而不是加强壁面本身的边界条件。这个所谓的动力壁面边界条件导致了壁面附近表面曲率的调整。

如果 θ_w 是壁面的接触角，那么挨着壁面的实际单元的表面法向为

$$\hat{n} = \hat{n}_w \cos\theta_w + \hat{t}_w \sin\theta_w \tag{5-18}$$

这里 $\hat{n}_w$ 和 $\hat{t}_w$ 分别是壁面的单位法向量和切向量。这个接触角与一个单元正常计算的表面法向远离壁面的联合决定了表面的局部曲率，这个曲率常用于调整表面张力计算中的体积力项。

接触角 θ_w 壁面和壁面上界面切线的夹角，由 Wall 对话框列表中 Wall Adhesion 中的第一相里面度量，如图 5-2 所示。

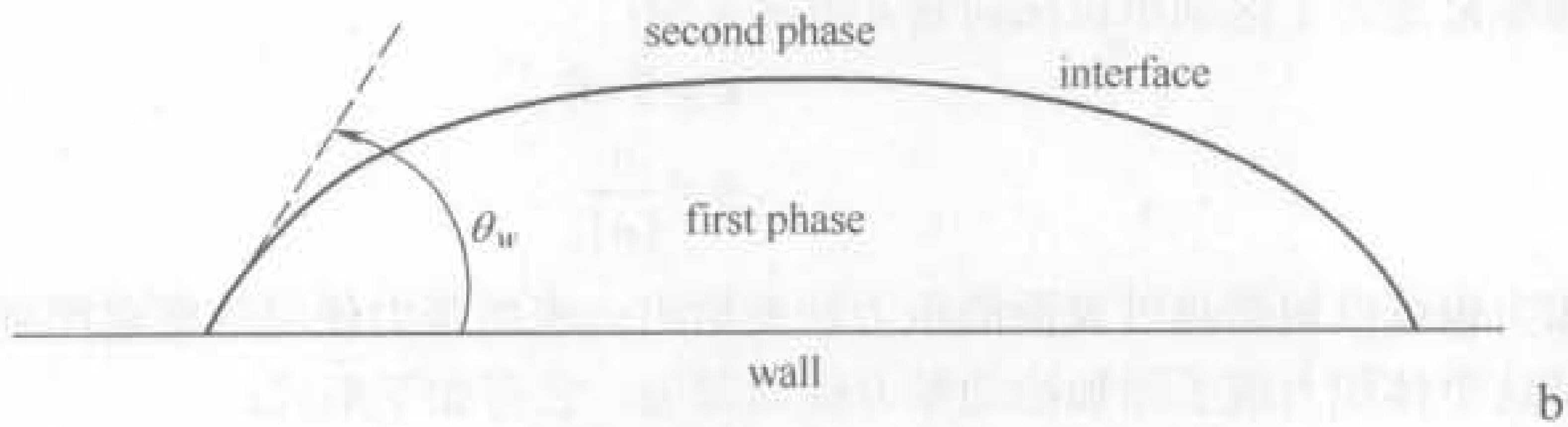


图 5-2 接触角

5.2 混合模型

5.2.1 混合模型概述及其局限

混合模型(mixture model)是一种简化的多相流模型，它用于模拟各相有不同速度的多相流，但是假定了在短空间尺度上局部的平衡。相之间的耦合应当是很强的，它也用于模拟有强烈耦合的各向同性多相流和各相以相同速度运动的多相流。

混合模型可以模拟 n 相(流体或粒子)流，通过求解混合相的连续性、动量和能量方程，第二相的体积分数方程，以及相对速度的代数实现。典型的应用包括沉降(sedimentation)、旋风分离器(cyclone separators)、低载荷粒子流(particle-laden flow with low loading)，以及气相体积分数很低的泡状流。

混合模型是欧拉模型在几种情形下的很好替代：①当存在大范围的颗粒相分布或者界面的规律未知或者它们的可靠性有疑问时，完善的多相流模型是不现实的；②当求解变量的个数小于完善的多相流模型时，像混合模型这样简单的模型能和完善的多相流模型一样取得好的结果。

在 Fluent 中使用混合模型存在以下一些限制。

- 必须使用分离求解器。
- 只有一相是可压缩的。

- 周期流动(比质量流率或比压降)问题不能和混合模型同时计算。
- 组分混合和反应流动问题不能和混合模型同时计算。
- 凝固与溶化模型不能与混合模型同时使用。
- 大涡湍流模型不能使用在混合模型中。
- 二阶隐式时间步算法不能用在混合模型中。
- 混合模型不能用于无粘流。
- 壁面壳传导模型不能和 VOF 模型同时计算。

5.2.2 控制方程

与 VOF 模型一样，混合模型使用单流体方法。它有两方面不同于 VOF 模型。

- 混合模型允许相之间互相贯穿(interpenetrating)。所以对一个控制体积的体积分数 α_q 和 α_p 可以是 0 和 1 之间的任意值，取决于相 q 和相 p 所占有的空间。
- 混合模型使用了滑流速度的概念，允许相以不同的速度运动(注：相也可以假定以相同的速度运动，混合模型就简化为均匀多相流模型)。

混合模型求解混合相的连续性方程、混合的动量方程、混合的能量方程、第二相的体积分数方程，还有相对速度的代数表达(如果相以不同的速度运动)。

1. 混合模型的连续方程

混合模型的连续方程(continuity equation for the mixture)为

$$\frac{\partial}{\partial t}(\rho_m)+\nabla\cdot(\rho_m\vec{v}_m)=\dot{m} \tag{5-19}$$

式(5-19)中的 $\vec{v}_m$ 是质量平均速度，可用式(5-20)表示：

$$\vec{v}_m=\frac{\sum_{k=1}^{n}\alpha_k\rho_k\vec{v}_k}{\rho_m} \tag{5-20}$$

式(5-19)和式(5-20)中的 ρ_m 是混合密度，可用式(5-21)表示：

$$\rho_m=\sum_{k=1}^{n}\alpha_k\rho_k \tag{5-21}$$

式(5-20)和式(5-21)中的 α_k 是第 k 相的体积分数。式(5-19)中的 $\dot{m}$ 描述了由于气穴(详见 5.4 节)或用户定义的质量源的质量传递。

2. 混合模型的动量方程

混合模型的动量方程(momentum equation for the mixture)可以通过对所有相各自的动量方程求和来获得。它可表示为

$$\frac{\partial}{\partial t}(\rho_m\vec{v}_m)+\nabla\cdot(\rho_m\vec{v}_m\vec{v}_m)=-\nabla p+\nabla\cdot[\mu_m(\nabla\vec{v}_m+\nabla\vec{v}_m^T)]+$$

$$\rho_m\vec{g}+\vec{F}+\nabla\cdot(\sum_{k=1}^{n}\alpha_k\rho_k\vec{v}_{dr,k}\vec{v}_{dr,k}) \tag{5-22}$$

式中，n 是相数，$\vec{F}$ 是体积力，μ_m 是混合粘性：

$$\mu_m = \sum_{k=1}^{n} \alpha_k \mu_k \tag{5-23}$$

$\vec{v}_{dr,k}$ 是第二相 k 的飘移速度：

$$\vec{v}_{dr,k} = \vec{v}_k - \vec{v}_m \tag{5-24}$$

3. 混合模型的能量方程

混合模型的能量方程(Energy Equation for the Mixture)采用如下形式：

$$\frac{\partial}{\partial t}\sum_{k=1}^{n}(\alpha_k \rho_k E_k) + \nabla \cdot \sum_{k=1}^{n}[\alpha_k \vec{v}_k(\rho_k E_k + p)] = \nabla \cdot (k_{eff} \nabla T) + S_E \tag{5-25}$$

式中，k_{eff} 是有效热传导率($k + k_t$ ，这里 k_t 是湍流热传导率，根据使用的湍流模型定义)。右边的第一项代表了由于传导造成的能量传递。S_E 包含了所有的体积热源。

在方程(5-25)中，

$$E_k = h_k - \frac{p}{\rho_k} + \frac{v_k^2}{2} \tag{5-26}$$

对可压缩相；而 $E_k = h_k$ 是对不可压缩相的，这里 h_k 是第 k 相的显焓。

4. 相对(滑流)速度和漂移速度

相对速度(也指滑流速度)被定义为第二相(p)的速度相对于主相(q)的速度：

$$\vec{v}_{qp} = \vec{v}_p - \vec{v}_q \tag{5-27}$$

漂移速度和相对速度($\vec{v}_{qp}$)的关系通过式(5-28)表示：

$$\vec{v}_{dr,p} = \vec{v}_{qp} - \sum_{k=1}^{n} \frac{\alpha_k \rho_k}{\rho_m} \vec{v}_{qk} \tag{5-28}$$

Fluent 中的混合模型使用了代数滑移公式。代数滑移混合模型的基本假设是规定相对速度的代数关系，相之间的局部平衡应在短的空间长度标尺上达到。相对速度的形式由式(5-29)给出：

$$\vec{v}_{qp} = \tau_{qp} \vec{\alpha} \tag{5-29}$$

式中，$\vec{\alpha}$ 是第二相粒子的加速度，τ_{qp} 是粒子的弛豫时间。根据 Manninen 理论，τ_{qp} 的形式为

$$\tau_{qp} = \frac{(\rho_m - \rho_p) d_p^2}{18 \mu_q f_{drag}} \tag{5-30}$$

式中，d_p 是第二相颗粒(或液滴或气泡)的直径，曳力函数 f_{drag} 来自 Schiller 和 Naumann 的研究：

$$f_{drag} = \begin{cases} 1 + 0.15 Re^{0.687} & Re \leqslant 1000 \\ 0.0183 Re & Re > 1000 \end{cases} \tag{5-31}$$

加速度 $\vec{\alpha}$ 的形式为

$$\vec{\alpha} = \vec{g} - (\vec{v}_m \cdot \nabla \vec{v}_m) - \frac{\partial \vec{v}_m}{\partial t} \tag{5-32}$$

最简单的代数滑移公式是所谓的漂移流量模型，其中粒子的加速度由重力或离心力给出，粒子的弛豫时间考虑其他粒子的存在而被修正。

注意：如果不需求解滑移速度，混合模型就简化成了均匀多相流模型。除此之外，混合模型还可以为滑移速度使用其他代数滑移方法让用户定制化(用户定义函数)。详细内容见 Fluent 的 UDF 手册。

5. 第二相的体积分数方程

从第二相 p 的连续方程，可以得到第二相 p 的体积分数方程为

$$\frac{\partial}{\partial t}(\alpha_p\rho_p)+\nabla\cdot(\alpha_p\rho_p\vec{v}_m)=-\nabla\cdot(\alpha_p\rho_p\vec{v}_{dr,p}) \tag{5-33}$$

5.3 欧拉模型

5.3.1 欧拉模型概述及其局限

采用欧拉模型，第二相的数量仅仅因为内存要求和收敛行为而受到限制，只要有足够的内存，任何数量的第二相都可以模拟。然而，对于复杂的多相流流动，会发现解由于收敛性而受到限制。

Fluent 6.0 以后的欧拉多相流模型不同于 Fluent 4 中的欧拉模型，在 Fluent 4 中液-液和液-固(granular)多相流动没有全局的差别。颗粒流是一种简单的流动，它涉及至少有一相被指定为颗粒相。

Fluent 6.0 以后的欧拉模型解是基于以下的假设。

- 单一的压力是被各相共享的。
- 动量和连续性方程是对每一相求解。

下面的参数对颗粒相是有效的。

- 颗粒温度(固体波动的能量)是对每一固体相计算的。这是基于代数关系的。
- 固体相的剪切和可视粘性是把分子运动论用于颗粒流而获得的。摩擦粘性也是有效的。
- 几个相间的曳力系数函数是有效的，它们适合于不同类型的多相流系(也可以通过用户定义函数修改相间的曳力系数，详细描述见 UDF 手册)。
- 所有的紊流模型都是有效的，可以用于所有相或者混合相。

在 Fluent 中欧拉多相流模型受到以下限制。

- 只有 κ-ε 模型能用于紊流。
- 颗粒跟踪(使用 Lagrangian 分散相模型)仅与主相相互作用。
- 周期流动(比质量流率或比压降)问题不能和欧拉模型同时计算。
- 压缩流动是不允许的。
- 无粘流是不允许的。
- 二阶隐式时间步算法不能用在欧拉模型中。
- 组分混合和反应流动模型不能使用。
- 模型中不能考虑热传输。
- 相间的质量传输仅允许在气穴问题中，而在蒸发和压缩中是不行的。

关于稳定性和收敛性(stability and convergence)，说明如下。

求解多相流系统的过程本来是困难的，会遇到稳定性和收敛性的问题，尽管现在的算法比 Fluent 4 中用得更稳定了。如果要求解时间依赖问题，并且流体预充填(patched fields)用于初始条件，建议采用较小的时间步长迭代几步，至少要比流动的特性时间小一个数量级。在迭代几步后可以增加时间步长的大小。对稳态问题建议开始时为体积分数用较小的欠松弛因子。

5.3.2 控制方程

单相模型中，只求解一套动量和连续性的守恒方程，为了实现从单相模型到多相模型的改变，必须引入附加的守恒方程。在引入附加的守恒方程的过程中，必须修改原始的设置。这个修改涉及多相体积分数 $\alpha_1,\alpha_2,\cdots,\alpha_n$ 的引入和相之间动量交换的机理。

1. 体积分数

作为互相贯穿且连续的多相流动的描述构成了相体积分数(volume fractions)的概念，这里表示为 α_q 。体积分数代表了每相所占据的空间，并且每相独自地满足质量和动量守恒定律。守恒方程的获得可以使用全体平均每一相的局部瞬态平衡或者使用混合理论方法。

q 相的体积 V_q 定义为：

$$V_q = \int_V \alpha_q \mathrm{d}V \tag{5-34}$$

式(5-34)中的 α_q 有下式成立：

$$\sum_{q=1}^{n} \alpha_q = 1 \tag{5-35}$$

q 相的有效密度为

$$\hat{\rho}_q = \alpha_q \rho_q \tag{5-36}$$

式中：ρ_q 是 q 相的物理密度。

2. 守恒方程

欧拉模型的守恒方程(conservation equations)也是由质量守恒方程(连续性方程)、动量守恒方程、能量守恒方程组成，详细描述如下。

1) 质量守恒方程

q 相的连续方程为

$$\frac{\partial}{\partial t}(\alpha_q \rho_q) + \nabla \cdot (\alpha_q \rho_q v_q) = \sum_{q=1}^{n} \dot{m}_{pq} \tag{5-37}$$

式中：v_q 是 q 相的速度，$\dot{m}_{pq}$ 表示了从第 p 相到 q 相的质量传递。从质量守恒方程可得

$$\dot{m}_{pq} = -\dot{m}_{qp} \tag{5-38}$$

和

$$\dot{m}_{pp} = 0 \tag{5-39}$$

2) 动量守恒方程

q 相的动量平衡产生了

$$\frac{\partial}{\partial t}(\alpha_q\rho_q\vec{v}_q)+\nabla\cdot(\alpha_q\rho_q\vec{v}_q\vec{v}_q)=-\alpha_q\nabla p+\nabla\cdot\overline{\overline{\tau}}_q+\sum_{p=1}^{n}(\vec{R}_{pq}+\dot{m}_{pq}\vec{v}_{pq})+\alpha_q\rho_q(\vec{F}_q+\vec{F}_{\text{lift},q}+\vec{F}_{\text{vm},q}) \tag{5-40}$$

式中，$\overline{\overline{\tau}}_q$ 是第 q 相的压力应变张量(stress-strain tensor)：

$$\overline{\overline{\tau}}_q=\alpha_q\mu_q(\nabla\vec{v}_q+\nabla\vec{v}_q^T)+\alpha_q\left(\lambda_q-\frac{2}{3}\mu_q\right)\nabla\cdot\vec{v}_q\overline{\overline{I}} \tag{5-41}$$

式中：μ_q 和 λ_q 是 q 相的剪切和体积粘度，$\vec{F}_q$ 是外部体积力，$\vec{F}_{\text{lift},q}$ 是升力，$\vec{F}_{\text{vm},q}$ 是虚拟质量力，$\vec{R}_{pq}$ 是相之间的相互作用力，p 是所有相共享的压力。$\vec{v}_{pq}$ 是相间的速度，定义如下：如果 $\dot{m}_{pq}>0$ (也就是，相 p 的质量传递到相 q)，$\vec{v}_{pq}=\vec{v}_p$；如果 $\dot{m}_{pq}<0$ (也就是，相 q 的质量传递到相 p)，$\vec{v}_{pq}=\vec{v}_q$ 和 $\vec{v}_{pq}=\vec{v}_{qp}$。

方程(5-39)必须有合适的表达使相间作用力 $\vec{R}_{pq}$ 封闭。这个力依赖于摩擦、压力、内聚力和其他影响，并服从条件 $\vec{R}_{pq}=-\vec{R}_{qp}$ 和 $\vec{R}_{pp}=0$。

Fluent 使用下面形式的相互作用力：

$$\sum_{p=1}^{n}\vec{R}_{pq}=\sum_{p=1}^{n}K_{pq}(\vec{v}_p-\vec{v}_q) \tag{5-42}$$

式中，$K_{pq}(=K_{qp})$ 是相间动量交换系数。

3) 升力

对多相流动，Fluent 能包含第二相粒子(或液滴或气泡)的升力的影响。这些升力作用于粒子主要是由于主相流场的速度梯度。对大的粒子，升力更重要，但是 Fluent 的模型假定粒子的直径远小于粒子间的距离。这样，对紧密充填的颗粒和非常小的粒子包含升力就不合适了。

主相 q 中作用于第二相 p 的升力由下式计算：

$$F_{\text{lift}}=-0.5\rho_q\alpha_p\left|\vec{v}_q-\vec{v}_p\right|\times(\nabla\times\vec{v}_q) \tag{5-43}$$

升力 F_{lift} 将会为两相添加到动量方程的右边($F_{\text{lift},q}=-F_{\text{lift},p}$)。

大多数情形下，升力相对于曳力是不重要的，因此不必要包含这个额外的项。如果升力是重要的(例如，如果相分离很快)，包含这项是合适的。默认情况，F_{lift} 是不包含的。如果需要，升力和升力系数应为每一对相指定。

4) 虚拟质量力

对多相流动，当第二相 q 相对于主相 q 加速时，Fluent 包含虚拟质量的影响。主相质量的惯性遇到加速的粒子(或液滴或气泡)对粒子施加一个虚拟质量力：

$$F_{\text{vm}}=0.5\alpha_p\rho_q\left(\frac{\mathrm{d}qv_q}{\mathrm{d}t}-\frac{\mathrm{d}pv_p}{\mathrm{d}t}\right) \tag{5-44}$$

$\frac{\mathrm{d}q}{\mathrm{d}t}$ 相表示从式(5-45)中派生出来的相物质时间：

$$\frac{\mathrm{d}q(\phi)}{\mathrm{d}t}=\frac{\partial(\phi)}{\partial t}+(\vec{v}_q\cdot\nabla)\phi \tag{5-45}$$

虚拟质量力 F_{vm} 将会为两相添加到动量方程的右边($F_{vm,q}=-F_{vm,p}$)。

当第二相的密度远小于主相的密度时，虚拟质量影响是重要的(例如对于瞬时的气泡柱)。默认情况， F_{vm} 是不包含的。

5) Fluent 求解的具体方程

Fluent 求解的液-液和颗粒多相流动的方程，列举如下作为 n 相流动的一般情形。

(1) 连续方程

每相的体积分数从连续方程计算：

$$\frac{\partial}{\partial t}(\alpha_q)+\nabla\cdot(\alpha_q\bar{v}_q)=\frac{1}{\rho_q}\left(\sum_{p=1}^{n}\dot{m}_{pq}-\alpha_q\frac{\mathrm{d}q\rho_q}{\mathrm{d}t}\right) \tag{5-46}$$

对每个第二相的连续方程的解，以及体积分数的和为 1 的条件(由方程(5-34)给出)，可以作为主相体积分数代入这个方程计算。这种处理对液-液和颗粒流动是公用的。

(2) 液-液动量方程

流体相 q 的动量守恒方程为

$$\begin{aligned}\frac{\partial}{\partial t}(\alpha_q\rho_q\bar{v}_q)+\nabla\cdot(\alpha_q\rho_q\bar{v}_q\bar{v}_q)=&-\alpha_q\nabla p+\nabla\cdot\bar{\bar{\tau}}_q+\alpha_q\rho_q\bar{g}+\\&\alpha_q\rho_q(\bar{F}_q+\bar{F}_{\mathrm{lift},q}+\bar{F}_{\mathrm{vm},q})+\sum_{p=1}^{n}[K_{pq}(\bar{v}_p-\bar{v}_q)+\dot{m}_{pq}\bar{v}_{pq}]\end{aligned} \tag{5-47}$$

式中： $\bar{g}$ 表示由于重力的重力加速度， $\bar{F}_q$、$\bar{F}_{\mathrm{lift},q}$ 和 $\bar{F}_{\mathrm{vm},1}$ 的定义见方程(5-40)。

(3) 液-固动量方程

Fluent 使用多流体粒子模型(multi-fluid granular model)来描述液体-固体的混合行为。固体相应力来自颗粒碰撞产生的随机粒子运动和气体分子的热扩散之间的类比，并考虑了颗粒相无伸缩性。正如气体的情形，颗粒速度波动的强度决定了应力、粘度和固相的压力。与颗粒速度相关的动能被假想热能(pseudothermal)或者与粒子随机运动平方成比例的颗粒温度所描绘。

液体相的动量守恒方程相似于方程(5-46)，第 s 固体相的方程为

$$\begin{aligned}\frac{\partial}{\partial t}(\alpha_s\rho_s\bar{v}_s)+\nabla\cdot(\alpha_s\rho_s\bar{v}_s\bar{v}_s)=&-\alpha_s\nabla p-\nabla p_s+\nabla\cdot\bar{\bar{\tau}}_s+\alpha_s\rho_s\bar{g}+\\&\alpha_s\rho_s(\bar{F}_s+\bar{F}_{\mathrm{lift},s}+\bar{F}_{\mathrm{vm},s})+\sum_{l=1}^{n}[K_{ls}(\bar{v}_l-\bar{v}_s)+\dot{m}_{ls}\bar{v}_{ls}]\end{aligned} \tag{5-48}$$

式中： p_s 是第 s 固相压力， $K_{ls}=K_{sl}$ 是液体或固体相 l 和固体相 s 之间的动量交换系数， n 为相的总数， $\bar{F}_q$、$\bar{F}_{\mathrm{lift},q}$ 和 $\bar{F}_{\mathrm{vm},q}$ 的定义见式(5-40)。

3. 相间交换系数

从方程(5-47)和方程(5-48)可以看出相之间的动量交换是以液-液交换系数 K_{pq} 的值为基础的，对颗粒流动，液-固和固-固交换系数为 K_{ls} 。

1) 液-液交换系数

对液-液流动，每个第二相被假定为液滴或气泡的形式。如何把流体中的一相指定为颗粒相很重要。例如，流动中有不同数量的两种流体，起支配作用的流体应作为主要流体，由于稀少的流体更可能形成液滴或气泡。这些气泡，液-液或气-液混合类型的交换系数可

以写成以下通用形式：

$$K_{pq} = \frac{\alpha_p \rho_p f}{\tau_p} \tag{5-49}$$

式中，曳力函数 f 对不同的交换系数模型定义不同(如下面的描述)，颗粒弛豫时间 τ_p 定义为

$$\tau_p = \frac{\rho_p d_p^2}{18\mu_q} \tag{5-50}$$

式中，d_p 是 p 相液滴或气泡的直径。

几乎所有 f 的定义都包含一个基于相对雷诺数(Re)的曳力系数(C_D)。这个曳力系数在不同的交换系数模型中是不同的。

(1) Schiller and Naumann 模型

$$f = \frac{C_D Re}{24} \tag{5-51}$$

式中的 C_D 由下式确定：

$$C_D = \begin{cases} 24(1+0.15Re^{0.687})/Re & Re \leqslant 1000 \\ 0.44 & Re > 1000 \end{cases} \tag{5-52}$$

式(5-51)和式(5-52)中的 Re 是相对雷诺数，主相 q 和第二相 p 的相对雷诺数从式(5-53)获得：

$$Re = \frac{\rho_q \left|\vec{v}_p - \vec{v}_q\right| d_p}{\mu_q} \tag{5-53}$$

第二相 p 和 r 的相对雷诺数从式(5-54)获得：

$$Re = \frac{\rho_{rp} \left|\vec{v}_r - \vec{v}_p\right| d_{rp}}{\mu_{rp}} \tag{5-54}$$

式中：$\mu_{rp} = \alpha_p \mu_p + \alpha_r \mu_r$ 是相 p 和 r 的混合速度。

(2) Morsi and Alexander 模型

$$f = \frac{C_D Re}{24} \tag{5-55}$$

式中的 C_D 由下式确定：

$$C_D = \alpha_1 + \frac{\alpha_2}{Re} + \frac{\alpha_3}{Re^2} \tag{5-56}$$

式(5-55)和式(5-56)中的 Re 数由方程(5-53)和式(5-54)定义。α_1、α_2、α_3 定义如下：

$$\alpha_1,\alpha_2,\alpha_3 = \begin{cases} 0, 18, 0 & 0 < Re < 0.1 \\ 3.690, 22.73, 0.0903 & 0.1 < Re < 1 \\ 1.222, 29.1677, -3.8889 & 1 < Re < 10 \\ 0.6167, 46.50, -116.67 & 10 < Re < 100 \\ 0.3644, 98.33, -2778 & 100 < Re < 1000 \\ 0.357, 148.62, -47500 & 1000 < Re < 5000 \\ 0.46, -490.546, 578700 & 5000 < Re < 10000 \\ 0.5191, -1662.5, 5416700 & Re > 10000 \end{cases} \tag{5-57}$$

Morsi and Alexander 模型是最完善的，频繁地在雷诺数的大范围内调整函数定义，但是采用这个模型比其他模型更不稳定。

(3) 对称模型

$$K_{pq}=\frac{\alpha_p(\alpha_p\rho_p+\alpha_q\rho_q)f}{\tau_{pq}} \tag{5-58}$$

式中的 τ_{pq} 和 f 由以下两式确定：

$$\tau_{pq}=\frac{(\alpha_p\rho_p+\alpha_q\rho_q)\left(\frac{d_p+d_q}{2}\right)^2}{18(\alpha_p\mu_p+\alpha_q\mu_q)} \tag{5-59}$$

$$f=\frac{C_D Re}{24} \tag{5-60}$$

式(5-60)中的 C_D 由下式确定：

$$C_D=\begin{cases}24(1+0.15Re^{0.687})/Re & Re\leqslant 1000\\ 0.44 & Re>1000\end{cases} \tag{5-61}$$

式(5-60)和式(5-61)中的 Re 数由式(5-53)和式(5-54)定义。

在流动中，区域内的某个地方的第二相(分散相)在另一个区域变成主相(连续相)。例如，如果空气注入充满一半水的容器的底部，在容器的底半部空气是分散相，在容器的顶半部空气是连续相。这个模型也用于两相之间的相互作用。

在该模型中，可以为每一对相指定不同的交换系数，也可以为每一对相使用用户定义函数定义交换系数。如果交换系数等于零(也就是，交换系数没有指定)，流体的流动区域将会独立地计算，并使用这个唯一的相互作用作为每个计算单元内各相补充的体积分数。

2) 液体-固体交换系数

液体-固体的交换系数 K_{sl} 以下面的通用形式写出：

$$K_{sl}=\frac{\alpha_s\rho_s f}{\tau_s} \tag{5-62}$$

式中，f 对不同的交换系数模型(如下描述)定义不同。颗粒的弛豫时间 τ_s 定义为

$$\tau_s=\frac{\rho_s d_s^2}{18\mu_l} \tag{5-63}$$

式中，d_s 是 s 相颗粒的直径。

所有 f 的定义都包含基于相对雷诺数的曳力函数，这个曳力函数在不同的交换系数模型中是不同的。

(1) Syamlal-O'Brien 模型

$$f=\frac{C_D Re_s\alpha_l}{24v_{r,s}^2} \tag{5-64}$$

这里曳力函数采用由 Dalla Valle 给出的形式：

$$C_D=\left(0.63+\frac{4.8}{\sqrt{Re_s/v_{r,s}}}\right)^2 \tag{5-65}$$

这个模型是基于流化床或沉淀床颗粒的末端速度的测量，并使用了体积分数和相对雷诺数的函数关系式：

$$Re_s = \frac{\rho_l d_s |\vec{v}_s - \vec{v}_l|}{\mu_l} \tag{5-66}$$

式中，下标 l 是第 l 液体相，s 是第 s 固体相，d_s 是第 s 固体相颗粒的直径。

这样，液体–固体交换系数有如下形式：

$$K_{sl} = \frac{3\alpha_s \alpha_l \rho_l}{4v_{r,s}^2 d_s} C_D \left(\frac{Re_s}{v_{r,s}}\right) |\vec{v}_s - \vec{v}_l| \tag{5-67}$$

式中，$v_{r,s}$ 是与固体相相关的末端速度：

$$v_{r,s} = 0.5(A - 0.06Re_s + \sqrt{(0.06Re_s)^2 + 0.12Re_s(2B - A) + A^2}) \tag{5-68}$$

其中，

$$A = \alpha_l^{4.14} \tag{5-69}$$

对 $\alpha_l \leqslant 0.85$，

$$B = 0.8\alpha_l^{1.28} \tag{5-70}$$

对 $\alpha > 0.85$，

$$B = \alpha_l^{2.65} \tag{5-71}$$

这个模型在固体相的剪切应力根据 Syamlal 等定义时(参见式(5-83))比较合适。

(2) Wen 和 Yu 模型

在该模型下，液体–固体交换系数有如下形式：

$$K_{sl} = \frac{3}{4} C_D \frac{\alpha_s \alpha_l \rho_l |\vec{v}_s - \vec{v}_l|}{d_s} \alpha_l^{-2.65} \tag{5-72}$$

式中的 C_D 由下式确定：

$$C_D = \frac{24}{\alpha_l Re_s}[1 + 0.15(\alpha_l Re_s)^{0.687}] \tag{5-73}$$

式中，Re 数由方程(5-66)定义。

这个模型适合于稀释系统。

(3) Gidaspow 模型

该模型是 Wen and Yu 模型和 Ergun 方程的联合。当 $\alpha_l > 0.8$ 时，液体–固体交换系数 K_{sl} 有如下形式：

$$K_{sl} = \frac{3}{4} C_D \frac{\alpha_s \alpha_l \rho_l |\vec{v}_s - \vec{v}_l|}{d_s} \alpha_l^{-2.65} \tag{5-74}$$

式中的 C_D 由下式确定：

$$C_D = \frac{24}{\alpha_l Re_s}[1 + 0.15(\alpha_l Re_s)^{0.687}] \tag{5-75}$$

当 $\alpha_l \leqslant 0.8$ 时，

$$K_{sl} = 150\frac{\alpha_s(1-\alpha_l)\mu_l}{\alpha_l d_s^2} + 1.75\frac{\rho_l \alpha_s |\vec{v}_s - \vec{v}_l|}{d_s} \tag{5-76}$$

这个模型适合密集的流化床计算。

3) 固体-固体交换系数

固体-固体交换系数 K_{ls} 有如下形式：

$$K_{sl}=\frac{3(1+e_{ls})\left(\frac{\pi}{2}+C_{\mathrm{fr},ls}\frac{\pi^2}{8}\right)\alpha_s\rho_s\alpha_l\rho_l(d_l+d_s)^2 g_{0,ls}}{2\pi(\rho_l d_l^3+\rho_s d_s^3)}\left|\vec{v}_l-\vec{v}_s\right| \tag{5-77}$$

式中：e_{ls} 为归还系数(在下一小节中描述)；$C_{\mathrm{fr},ls}$ 为第 l 和第 s 相之间的摩擦系数，固体相颗粒为 $C_{\mathrm{fr},ls}=0$；d_l 为固体 l 颗粒的直径；$g_{0,ls}$ 为径向分布系数(在下一小节中描述)。

4. 固体压力

对可压缩机制下的颗粒流动(也就是，固体的体积分数小于允许的最大值的地方)，固体压力(solids pressure)独立计算，并且用作颗粒相动量方程中的压力梯度相 ∇p_s。因为 Maxwellian 速度分布用于颗粒，颗粒温度引入了模型，并出现在固体压力和粘度的表达式中。由于颗粒的碰撞，固体压力由动能项和第二相组成：

$$p_s=\alpha_s\rho_s\Theta_s+2\rho_s(1+e_{ss})\alpha_s^2 g_{0,ss}\Theta_s \tag{5-78}$$

式中：e_{ss} 是颗粒碰撞的归还系数，$g_{0,ss}$ 是径向分布函数，Θ_s 是颗粒温度。在 Fluent 中，e_{ss} 默认值是 0.9，但是这个值能调整以适合颗粒类型。颗粒温度 Θ_s 是与颗粒运动的波动动能成比例的，将在本部分的后面描述。函数 $g_{0,ss}$ (更详细的内容下面描述)是分布函数，这个函数控制了从 $\alpha<\alpha_{s,\max}$ (这里固体颗粒之间的距离可以继续减小)的可压缩条件到 $\alpha=\alpha_{s,\max}$ (这里距离的进一步减小不会发生)的不可压缩条件。$\alpha_{s,\max}$ 的默认值是 0.63，但是在问题设置过程中可以修改。

径向分布函数 g_0 是一个当固体颗粒相变密时用于修改颗粒之间碰撞概率的修正因子。这个函数也可解释为小球之间的无量纲距离：

$$g_0=\frac{s+d_p}{s} \tag{5-79}$$

式中，s 是颗粒之间的距离。从方程(5-79)可以观察出对稀疏固体相 $s\to\infty$，所以 $g_0\to 1$。当固体相紧凑到一定限制内，$s\to 0$，$g_0\to\infty$。径向分布函数与非均匀气体的 Chapman and Cowling's 理论的 χ 因子紧密联系。对稀有气体，χ 等于 1，当分子靠得非常近以致运动不可能发生时，它会逐渐增加并趋向无穷大。

径向分布函数没有统一的公式。Fluent 采用推荐的：

$$g_0=\left[1-\left(\frac{\alpha_s}{\alpha_{s,\max}}\right)^{\frac{1}{3}}\right]^{-1} \tag{5-80}$$

当固体相数大于 1 时，方程(5-80)扩展为

$$g_{0,ll}=\left[1-\left(\frac{\alpha_l}{\alpha_{l,\max}}\right)^{\frac{1}{3}}\right]^{-1} \tag{5-81}$$

式中，$\alpha_{l,\max}$ 是由你在问题的设置过程中指定的，并且

$$g_{0,lm}=\frac{d_m g_{0,ll}+d_l g_{0,mm}}{d_m+d_l} \tag{5-82}$$

5. 固体剪切应力

固体应力张量包含由于平移和碰撞从颗粒的动量交换中产生的剪切和体积粘性，粘性的摩擦分量也可以包含在当固体颗粒相达到最大固体颗粒分数时出现的粘塑性变迁中。

固体剪切粘度包括：碰撞和动力部分、摩擦部分，即

$$\mu_s=\mu_{s,\mathrm{col}}+\mu_{s,\mathrm{kin}}+\mu_{s,\mathrm{fr}} \tag{5-83}$$

1) 碰撞粘性(collisional viscosity)

剪切粘度的碰撞部分模化为

$$\mu_{s,\mathrm{col}}=\frac{4}{5}\alpha_s\rho_s d_s g_{0,ss}(1+\mathrm{e}_{ss})\left(\frac{\theta_s}{\pi}\right)^{1/2} \tag{5-84}$$

2) 动力粘度(kinetic viscosity)

Fluent 为动力部分提供了两种表达，默认的是 Syamlal et al 表达：

$$\mu_{s,\mathrm{kin}}=\frac{\alpha_s d_s\rho_s\sqrt{\theta_s\pi}}{6(3-\mathrm{e}_{ss})}\left[1+\frac{2}{5}(1+\mathrm{e}_{ss})(3\mathrm{e}_{ss}-1)\alpha_s g_{0,ss}\right] \tag{5-85}$$

也可以采用如下的 Gidaspow et al 表达：

$$\mu_{s,\mathrm{kin}}=\frac{10\rho_s d_s\sqrt{\theta_s\pi}}{96\alpha_s(1+\mathrm{e}_{ss})g_{0,ss}}\left[1+\frac{4}{5}g_{0,ss}\alpha_s(1+\mathrm{e}_{ss})\right]^2 \tag{5-86}$$

3) 体积粘度

固体体积粘度(bulk viscosity)解释为颗粒压缩和扩张的抵抗力。根据 Lun 等，有以下形式：

$$\lambda_s=\frac{4}{3}\alpha_s\rho_s d_s g_{0,ss}(1+\mathrm{e}_{ss})\left(\frac{\theta_s}{\pi}\right)^{1/2} \tag{5-87}$$

注：默认时，体积粘度被设置为常数 0。选择 Lun et al 表达或用户定义函数也是可能的。

4) 摩擦粘度

在低剪切密集流动中，固体的第二相体积分数接近于压缩极限，应力的产生主要是由于颗粒之间的摩擦。默认情况，由 Fluent 计算的固体剪切粘度不解释为颗粒之间的摩擦。

如果计算中包含摩擦粘度(frictional viscosity)，Fluent 使用 Schaeffer's 表达：

$$\mu_{s,\mathrm{fr}}=\frac{p_s\sin\phi}{2\sqrt{I_{2D}}} \tag{5-88}$$

式中：p_s 是固体压力；ϕ 是内部摩擦角；I_{2D} 是偏应力张量的第二不变式，它也可以被指定为常数或用户定义摩擦粘度。

6. 颗粒温度

第 s 固体相的颗粒温度(granular temperature)是与颗粒的随机运动的动能成比例的。从动能理论得到的输运方程采用如下形式[50]：

$$\frac{3}{2}\left[\frac{\partial}{\partial t}(\rho_s\alpha_s\Theta_s)+\nabla\cdot(\rho_s\alpha_s\vec{v}_s\Theta_s)\right]=(-p_s\overline{\overline{I}}+\overline{\overline{\tau}}_s):\nabla\vec{v}_s+\nabla\cdot(k_{\Theta_s}\nabla\Theta_s)-\gamma\Theta_s+\phi_{ls} \tag{5-89}$$

式中：$(-p_s\overline{\overline{I}}+\overline{\overline{\tau}}_s):\nabla\vec{v}_s$ 为固体应力张量产生的能量；$k_{\Theta_s}\nabla\Theta_s$ 为能量扩散(k_{Θ_s} 是扩散系数)；$\gamma\Theta_s$ 为能量的碰撞耗散；ϕ_{ls} 为第 l 相液体或固体相和第 s 固体相之间的能量交换。

方程(5-89)包含描述了颗粒能量扩散通量的 $k_{\Theta_s}\nabla\Theta_s$ 项。

能量的碰撞耗散 $\gamma\Theta_s$ 代表了由于颗粒之间的碰撞在第 s 固体相内的能量耗散率。这项也可以由 Lun 等得来的表达描述：

$$\gamma_{\Theta m}=\frac{12(1-e_{ss}^2)g_{0,ss}}{d_s\sqrt{\pi}}\rho_s\alpha_s^2\Theta_s^{3/2} \tag{5-90}$$

从第 s 固体相到第 l 液体或固体相粒子速度的随机波动动能的传递由 ϕ_{ls} 描述：

$$\phi_{ls}=-3K_{ls}\Theta_s \tag{5-91}$$

Fluent 当前使用颗粒温度的代数关系，这可以通过忽略输运方程中的对流和扩散获得方程(5-89)。

7. 湍流模型

为了描述单相中速度及标量的湍流、波动的影响，Fluent 使用了不同类型的封闭模型。与单相流动相比，多相流动动量方程中所模拟的项数是非常大的，这使得多相流模拟中的湍流模型非常复杂。

在 k-ε 模型内 Fluent 提供了 3 种方法模拟多相流中的湍流。

- 混合湍流模型(默认)。
- 分散湍流模型。
- 分相湍流模型。

模型的选择依赖于应用中第二相湍流的重要性。

注意： 下面给出的每一种方法的描述都是基于标准 k-ε 模型。将多相流的湍流修正为 RNG 和 realizable k-ε 模型时，作相似地处理，因此这里不再明确地给出。

1) 混合湍流模型

混合湍流模型(mixture turbulence model)是默认的多相湍流模型，它代表了单相 k-ε 模型的第一扩展，它应用于相分离、分层(或接近分层)的多相流，以及相之间的密度比接近 1 的湍流状况。在这种情形下，使用混合属性和混合速度捕获湍流的重要特征是足够的。

描述这个模型的 k-ε 方程如下：

$$\frac{\partial}{\partial t}(\rho_m k)+\nabla\cdot(\rho_m\vec{v}_m k)=\nabla\cdot\left(\frac{\mu_{t,m}}{\sigma_k}\nabla k\right)+G_{k,m}-\rho_m\varepsilon \tag{5-92}$$

和

$$\frac{\partial}{\partial t}(\rho_m\varepsilon)+\nabla\cdot(\rho_m\vec{v}_m\varepsilon)=\nabla\cdot\left(\frac{\mu_{t,m}}{\sigma_\varepsilon}\nabla\varepsilon\right)+\frac{\varepsilon}{k}(C_{1\varepsilon}G_{k,m}-C_{2\varepsilon}\rho_m\varepsilon) \tag{5-93}$$

以上两式中的混合密度 ρ_m 和混合速度 $\vec{v}_m$ 通过式(5-94)和式 5-95 计算：

$$\rho_m = \sum_{i=1}^{N} \alpha_i \rho_i \tag{5-94}$$

$$\vec{v}_m = \frac{\sum_{i=1}^{N} \alpha_i \rho_i \vec{v}_i}{\sum_{i=1}^{N} \alpha_i \rho_i} \tag{5-95}$$

式(5-92)和式(5-93)中的湍流粘度 $\mu_{t,m}$ 根据式(5-96)计算：

$$\mu_{t,m} = \rho_m C_\mu \frac{k^2}{\varepsilon} \tag{5-96}$$

式(5-92)和式(5-93)中的湍流动能的产生 $G_{k,m}$ 由式(5-97)计算：

$$G_{k,m} = \mu_{t,m}[\nabla \vec{v}_m + (\nabla \vec{v}_m)^T] : \nabla \vec{v}_m \tag{5-97}$$

这些方程中的常数与单相 k-ε 模型的描述相同。

2)　分散湍流模型

当第二相的浓度稀时，分散湍流模型(dispersed turbulence model)是合适的模型。这种情形下，颗粒间的碰撞可忽略，而对第二相随机运动起支配作用的是主相湍流的影响。所以第二相的波动量根据主相的平均特征和颗粒弛豫时间和粒子相互作用时间的旋涡给出。

当明显地有一个主连续相和其他的是分散稀释的第二相时，这个模型是比较合适的。在计算中，Fluent 中模拟湍流的分散方法涉及以下假设。

- 对连续相修正 k-ε 模型：连续相湍流预测是使用标准 k-ε 模型并补充包含相间湍流动量传递的附加项获得的。
- 对离散相用 Tchen-theory 关系：分散相湍流量度的预测是使用均匀湍流离散粒子的 Tchen-theory 传播理论获得的。
- 相间湍流动量传递：在湍流多相流动中，动量交换项包含了分散相瞬态分布和湍流流体运动之间的关系，考虑通过湍流流体运动输送分散相的传播是可能的。
- 相加权平均方法：在模拟湍流多相流的传播时其平均方法的选择是有影响的。两步平均法会导致出现相体积分数的波动。然而，当使用两步平均法结合对湍流的相加权平均时，体积分数的湍流波动不会出现。Fluent 使用相加权平均，因此没有体积分数的波动引入连续方程。

3)　连续相中的湍流

涡粘性模型常用于计算平均波动量。连续相 q 的雷诺应力张量采用如下形式：

$$\overline{\overline{\tau}}_q'' = -\frac{2}{3}(\rho_q k_q + \rho_q \mu_{t,q} \nabla \cdot \vec{U}_q)\overline{\overline{I}} + \rho_q \mu_{t,q}(\nabla \vec{U}_q + \nabla \vec{U}_q^T) \tag{5-98}$$

式中，$\vec{U}_q$ 是相加权速度。

根据 q 相的湍流动能，湍流粘度 $\mu_{t,q}$ 可由式(5-99)确定：

$$\mu_{t,q} = \rho_q C_\mu \frac{k_q^2}{\varepsilon_q} \tag{5-99}$$

载能湍流涡的特征时间定义如下：

$$\tau_{t,q}=\frac{3}{2}C_{\mu}\frac{k_q}{\varepsilon_q} \tag{5-100}$$

式中：ε_q 是耗散率，$C_{\mu}=0.09$。

湍流涡的长度标尺为

$$L_{t,q}=\sqrt{\frac{3}{2}}C_{\mu}\frac{k_q^{\frac{3}{2}}}{\varepsilon_q} \tag{5-101}$$

湍流预测从修正的 k - ε 模型获得：

$$\frac{\partial}{\partial t}(\alpha_q\rho_q k_q)+\nabla\cdot(\alpha_q\rho_q\vec{U}_q k_q)=\nabla\cdot\left(\alpha_q\frac{\mu_{t,q}}{\sigma_k}\nabla k_q\right)+\alpha_q G_{k,q}-\alpha_q\rho_q\varepsilon_q+\alpha_q\rho_q\Pi_{k_q} \tag{5-102}$$

和

$$\frac{\partial}{\partial t}(\alpha_q\rho_q\varepsilon_q)+\nabla\cdot(\alpha_q\rho_q\vec{U}_q\varepsilon_q)=\nabla\cdot\left(\alpha_q\frac{\mu_{t,q}}{\sigma_\varepsilon}\nabla\varepsilon_q\right)+\alpha_q\frac{\varepsilon_q}{k_q}(C_{1\varepsilon}G_{k,q}-C_{2\varepsilon}\rho_q\varepsilon_q)+\alpha_q\rho_q\Pi_{\varepsilon_q} \tag{5-103}$$

式中：Π_{k_q} 和 Π_{ε_q} 代表了分散相对连续相 q 的影响，$G_{k,q}$ 代表了湍流动能的产生。所有其他项与单相 k - ε 模型中的有相同的意义。

Π_{k_q} 项可从连续相的瞬态方程获得并采用如下形式，这里 M 代表第二相的数量：

$$\Pi_{k_q}=\sum_{p=1}^{M}\frac{K_{pq}}{\alpha_q\rho_q}\left[<\vec{v}_q^{\prime}\cdot\vec{v}_p^{\prime}>+(\vec{U}_p-\vec{U}_q)\cdot\vec{v}_{\mathrm{dr}}\right] \tag{5-104}$$

它可以简化为

$$\Pi_{k_q}=\sum_{p=1}^{M}\frac{K_{pq}}{\alpha_q\rho_q}(k_{pq}-2k_q+\vec{v}_{pq}\cdot\vec{v}_{\mathrm{dr}}) \tag{5-105}$$

式中：k_{pq} 是连续相 q 和离散相 p 的速度的协方差(从下面的方程(5-113)计算得)，$\vec{v}_{pq}$ 是相对速度，$\vec{v}_{\mathrm{dr}}$ 是漂移速度(由方程 5-118 定义)。

Π_{ε_q} 根据 Elgobashi et al 模化：

$$\Pi_{\varepsilon_q}=C_{3\varepsilon}\frac{\varepsilon_q}{k_q}\Pi_{k_q} \tag{5-106}$$

式中，$C_{3\varepsilon}=1.2$。

(1) 分散相中的湍流

表征运动的时间和长度标尺用于估计传播(dispersion)系数、相关函数和每一分散相的湍流动能。和作用于离散相 P 的惯性影响相连接的特征粒子弛豫时间定义为

$$\tau_{F,pq}=\alpha_p\rho_p K_{pq}^{-1}\left(\frac{\rho_p}{\rho_q}+C_v\right) \tag{5-107}$$

沿着颗粒轨道计算的 Lagrangian 积分时间标尺主要受交叉轨道的影响，定义为

$$\tau_{t,pq} = \frac{\tau_{t,q}}{\sqrt{(1+C_{\beta}\xi^2)}} \tag{5-108}$$

式中的ξ和C_β由式(5-109)和式(5-110)确定：

$$\xi = \frac{\left|\vec{v}_{pq}\right|\tau_{t,q}}{L_{t,q}} \tag{5-109}$$

$$C_\beta = 1.8 - 1.35\cos^2\theta \tag{5-110}$$

式中，θ是平均颗粒速度和平均相对速度的夹角。式(5-107)～式(5-108)中的两个特征时间比值记为

$$\eta_{pq} = \frac{\tau_{t,pq}}{\tau_{F,pq}} \tag{5-111}$$

根据 Simonin，Fluent 把离散相p的湍流量度写为如下形式：

$$k_p = k_q\left(\frac{b^2+\eta_{pq}}{1+\eta_{pq}}\right) \tag{5-112}$$

$$k_{pq} = 2k_q\left(\frac{b+\eta_{pq}}{1+\eta_{pq}}\right) \tag{5-113}$$

$$D_{t,pq} = \frac{1}{3}k_{pq}\tau_{t,pq} \tag{5-114}$$

$$D_p = D_{t,pq} + \left(\frac{2}{3}k_p - b\frac{1}{3}k_{pq}\right)\tau_{F,pq} \tag{5-115}$$

$$b = (1+C_V)\left(\frac{\rho_p}{\rho_q}+C_V\right)^{-1} \tag{5-116}$$

式中，$C_V = 0.5$是附加的质量系数。

(2)　相间湍流动量传递

多相流的湍流曳力项(方程(5-42)中的$K_{pq}(\vec{v}_p - \vec{v}_q)$)模化如下。对离散相$p$和连续相$q$：

$$K_{pq}(\vec{v}_p - \vec{v}_q) = K_{pq}(\vec{U}_p - \vec{U}_q) - K_{pq}\vec{v}_{dr} \tag{5-117}$$

式中，右边的第二项包含漂移速度：

$$\vec{v}_{dr} = -\left(\frac{D_p}{\sigma_{pq}\alpha_p}\nabla\alpha_p - \frac{D_q}{\sigma_{pq}\alpha_q}\nabla\alpha_q\right) \tag{5-118}$$

式中：D_p和D_q是扩散率，σ_{pq}是湍流 Schmidt 数。当在多相流中使用 Tchen 理论时，Fluent 假设$D_p = D_q = D_{t,pq}$，并默认σ_{pq}的值为 0.67。

在体积分数中漂移速度起因于湍流波动，当乘以交换系数K_{pq}时，它用作湍流动量交换项的修正。默认情况，不包含这个修正，但是在问题设置过程中，可以使用它。

(3)　每相的湍流模型

最普通的多相湍流模型为每一相求解一套k-ε输运方程。当湍流传递在相间起重要作用时，这个湍流模型是合适的选择。

注意：由于 Fluent 为每个第二相求解两个附加的输运方程，每相的湍流模型比分散相湍流模型大大地增加了计算的强度。

(4) 输运方程

雷诺应力张量和湍流粘度采用方程(5-98)和(5-99)计算。湍流预测从式(5-119)获得：

$$\frac{\partial}{\partial t}(\alpha_q\rho_q k_q)+\nabla\cdot(\alpha_q\rho_q\bar{U}_q k_q)=\nabla\cdot\left(\alpha_q\frac{\mu_{t,q}}{\sigma_k}\nabla k_q\right)+(\alpha_q G_{k,q}-\alpha_q\rho_q\varepsilon_q)+ \\ \sum_{l=1}^{N}K_{lq}(C_{lq}k_l-C_{ql}k_q)-\sum_{l=1}^{N}K_{lq}(\bar{U}_l-\bar{U}_q)\cdot\frac{\mu_{t,l}}{\alpha_l\sigma_l}\nabla\alpha_l+\sum_{l=1}^{N}K_{lq}(\bar{U}_l-\bar{U}_q)\cdot\frac{\mu_{t,q}}{\alpha_q\sigma_q}\nabla\alpha_q \tag{5-119}$$

和

$$\frac{\partial}{\partial t}(\alpha_q\rho_q\varepsilon_q)+\nabla\cdot(\alpha_q\rho_q\bar{U}_q\varepsilon_q)=\nabla\cdot\left(\alpha_q\frac{\mu_{t,q}}{\sigma_\varepsilon}\nabla\varepsilon_q\right)+\frac{\varepsilon_q}{k_q}(C_{1\varepsilon}\alpha_q G_{k,q}-C_{2\varepsilon}\alpha_q\rho_q\varepsilon_q+ \\ C_{3\varepsilon}\left[\sum_{l=1}^{N}K_{lq}(C_{lq}k_l-C_{ql}k_q)-\sum_{l=1}^{N}K_{lq}(\bar{U}_l-\bar{U}_q)\cdot\frac{\mu_{t,l}}{\alpha_l\sigma_l}\nabla\alpha_l+\sum_{l=1}^{N}K_{lq}(\bar{U}_l-\bar{U}_q)\cdot\frac{\mu_{t,q}}{\alpha_q\sigma_q}\nabla\alpha_q)\right] \tag{5-120}$$

C_{lq} 和 C_{ql} 可以近似地写为

$$C_{lq}=2 \\ C_{ql}=2\left(\frac{\eta_{lq}}{1+\eta_{lq}}\right) \tag{5-121}$$

式中，η_{lq} 由方程(5-111)定义(只是用 l 代替 p)。

(5) 相间湍流动量传递

湍流曳力项(方程(5-42)中的 $K_{pq}(\vec{v}_p-\vec{v}_q)$)模化如下。这里 l 是分散相(代替方程(5-42)中的 p)，q 是连续相：

$$\sum_{l=1}^{N}K_{lq}(\vec{v}_l-\vec{v}_q)=\sum_{l=1}^{N}K_{lq}(\bar{U}_l-\bar{U}_q)-\sum_{l=1}^{N}K_{lq}\vec{v}_{dr,lq} \tag{5-122}$$

式中：$\bar{U}_l$ 和 $\bar{U}_q$ 是相加权速度，$\vec{v}_{dr,lq}$ 是相 l 的漂移速度(使用方程(5-118)计算，用 l 取代 p)。注：Fluent 直接从输运方程计算扩散率 D_l 和 D_q，而不使用 Tchen 理论(一般它用于分散湍流模型)。

如上面指出的，漂移速度起因于体积分数的湍流波动。当乘以交换系数 K_{lq} 时，用于修正湍流中的动量交换项。默认情况，这个修正不包括，但是能在问题设置过程中使用它。

Fluent 中的每相的湍流模型解释为湍流场中一相对其他相的影响。如果想修正或增强多相湍流场之间的相互作用和相间湍流动量的传递，可以使用用户定义函数提供这些项。

8. Fluent 中的求解方法

对欧拉多相流计算，对压力速度耦合 Fluent 使用 Phase Coupled SIMPLE (PC-SIMPLE) 算法。PC-SIMPLE 是 SIMPLE 算法在多相流中的扩展。速度的求解被相耦合，但是用分离的方式。耦合求解器采用多重网格块代数方案来同时求解各相速度分量形成的矢量方程，然后，压力修正方程是基于总的体积连续而不是质量连续建立起来的。为了满足连续性的

约束压力和速度都被修正。

1)　压力修正方程

对不可压缩多相流，压力修正方程(the pressure-correction equation)采用如下形式：

$$\sum_{k=1}^{n}\left\{\frac{\partial}{\partial t}\alpha_k+\nabla\cdot\alpha_k\vec{v}_k'+\nabla\cdot\alpha_k\vec{v}_k^*-\frac{1}{\rho_k}\sum_{l=1}^{n}\dot{m}_{lk}\right\}=0 \tag{5-123}$$

式中，$\vec{v}_k'$ 是第 k 相的速度修正，$\vec{v}_k^*$ 是 $\vec{v}_k$ 的当前迭代值。速度修正本身表示为压力修正的函数。

2)　体积分数

体积分数(volume fractions)从相连续性方程获得。离散形式下，第 k 相体积分数方程为

$$\alpha_{p,k}\alpha_k=\sum_{nb}(\alpha_{nb,k}\alpha_{nb,k})+b_k=R_k \tag{5-124}$$

为了满足所有体积分数和为 1 的条件，还有

$$\sum_{k=1}^{n}\alpha_k=1 \tag{5-125}$$

5.4　气穴影响

当使用混合模型或欧拉模型时，两相流中可能会产生气穴，因此，Fluent 中也可以使用气穴的模型。

5.4.1　气穴模型概述及其局限

在恒定温度下对液体降压，可以降到饱和蒸汽压之下。在恒温下通过降低压力使液体破裂的过程称为气穴现象。液体包含微小空气泡或核，在降压下它们会长大并形成气穴。

当局部压力小于蒸发压力时，为两种互相贯穿的流体设计的气穴模型模拟了气泡的信息。气穴模型可以和混合模型(有无滑移速度)或欧拉多相模型一起使用。

气穴模型中作了如下假设。

- 计算中仅涉及两相的系统。
- 气泡既不创造也不破坏。
- 每个单位体积内气泡的数量是预先知道的。

Fluent 中应用气穴模型的局限如下。

- 气穴模型不能和 VOF 模型一起使用，因为 VOF 模型的表面跟踪方案和气穴模型的相互贯穿的连续性假设是不相容的。
- 气穴模型仅用于使用混合或欧拉模型的多相流模拟并且只能涉及两相。求解气穴模型使用没有滑移速度的混合模型总是可取的，如果求解的问题需要两相之间有重大的滑移，欧拉多相和滑移速度应打开。
- 使用气穴模型，只能第二相是可压缩的，主相必须是不可压缩的。

5.4.2 体积和气泡数量

单个气泡体积关于空间和时间的变化由式(5-126)给出：

$$\phi(\vec{r},t)=\frac{4}{3}\pi R^3 \tag{5-126}$$

式中，R 是气泡的半径。

蒸发的体积分数定义为

$$\alpha_v=\frac{\phi\eta}{1+\phi\eta} \tag{5-127}$$

式中，η 是单位流体体积内的气泡数量。

5.4.3 体积分数方程

体积分数方程(the volume fraction equation)从混合(m)的连续性方程获得。经过处理后，假定不可压缩的液体(l)，可以获得下面的表达式：

$$\frac{\partial}{\partial t}(\alpha_p)+\nabla(\alpha_p\vec{v}_m)=\frac{\rho_l}{\rho_m}\frac{\eta}{(1+\phi\eta)^2}\frac{\mathrm{d}\phi}{\mathrm{d}t}+\frac{\alpha\rho_v}{\rho_m}\frac{\mathrm{d}\rho_v}{\mathrm{d}t} \tag{5-128}$$

5.4.4 气泡动力学

由于气穴气泡在低温下形成液体，Fluent 等温模拟气穴流动，忽略了蒸发潜热。Rayleigh-Plesset 方程与压力和气泡体积 ϕ 相关：

$$R\frac{\mathrm{d}^2R}{\mathrm{d}t^2}+\frac{3}{2}\left(\frac{\mathrm{d}R}{\mathrm{d}t}\right)^2=\frac{p_B-p}{\rho_l}-\frac{2\sigma}{\rho_l R}-4\frac{\mu_l}{\rho_l R}\frac{\mathrm{d}R}{D\mathrm{d}t} \tag{5-129}$$

式中，p_B 表示气泡内的压力，由蒸汽的部分压力(p_v)和非凝结气体的部分压力(p)之和来描绘；σ 是表面张力系数。

为了简化计算，Fluent 假设气泡成长和破裂的过程由式(5-130)给出：

$$\frac{\mathrm{d}R}{\mathrm{d}t}=\begin{cases}\sqrt{\dfrac{2(p_B-p)}{3\rho_l}}, & p_v>p\\ -\sqrt{\dfrac{2(p_B-p)}{3\rho_l}}, & p_v<p\end{cases} \tag{5-130}$$

5.5 选择通用多相流模型

一般情况下，VOF 模型适合于分层的或自由表面流，而混合和欧拉模型适合于流动中有相混合或分离，或者分散相的体积分数超过 10%的情形。(流动中分散相的体积分数小于或等于 10%时可使用离散相模型)。

为了在混合模型和欧拉模型之间作出选择，除了前面的分析外，还应考虑以下几点。

- 如果分散相有着宽广的分布，混合模型是最可取的。如果分散相只集中在区域的

一部分，应当使用欧拉模型。

- 如果应用的系统的相间曳力规律是可利用的(在 Fluent 中，或通过 UDF 函数)，欧拉模型通常比混合模型能给出更精确的结果。如果相间的曳力规律不知道或者它们应用于你的系统是有疑问的，混合模型可能是更好的选择。
- 如果想解一个需要计算付出较少的简单的问题，混合模型可能是更好的选择，因为它比欧拉模型要少解一部分方程。如果精度比计算付出更重要，欧拉模型是更好的选择。但是请记住，复杂的欧拉模型比混合模型的计算稳定性要差。

对于欧拉模型，因计算复杂，稳定性较差，确定选用该模型后要考虑求解多相流问题的需求计算能力。要求的计算能力很强地依赖于所求解的输运方程的个数和耦合程度。对欧拉多相流模型，有大数量的高度耦合的输运方程，计算的耗费将很高，在设置问题前，应尽可能减少问题的求解量到最简化的可能形式。

在开始第一次求解尝试时，与其尽力去求解多相流动的所有的复杂方面，不如以简单近似的方面开始。简化多相流问题的一些建议列举如下。

- 使用六面体或四边形网格(而不用四面体或三角形网格)。
- 减少相的数目。

5.6　设置一般的多相流问题

5.6.1　使用一般多相流模型的步骤

设置和求解一般多相流问题的步骤的要点如下，随后的章节有各个子部分详细的讲述。这里给出的仅是与一般多相流计算相关的步骤。有关使用的其他模型和相关的多相流模型的输入的详细信息，将在这些模型中合适的部分给出。

(1) 选中想要使用的多相流模型(VOF、混合或欧拉)并指定相数。对于 VOF 模型，要指定 VOF 公式。

选择 Define→Models→Multiphase 命令。

(2) 从材料库中复制描述每相的材料。

选择 Define→Materials 命令。

如果使用的材料在库中没有，应创建一种新材料。

注意：　如果模型中含有微粒(granular)相，必须在 fluid materials category 中为它创建新材料(而不在 solid materials category 中)。

(3) 定义相，指定相间的相互作用(interaction)(例如，使用 VOF 模型时的表面张力(surface tension)，使用混合模型时的滑流速度，使用欧拉模型时的曳力函数)。

选择 Define→Phases 命令。

(4) (仅对欧拉模型)如果流动是紊流，定义多相紊流模型。

选择 Define→Models→Viscous 命令。

(5) 如果体积力存在，开启。

选择 Define→Operating Conditions 命令。

(6) 指定边界条件，包括第二相体积份额在流动边界和(如果在 VOF 模拟中模拟壁面附近)壁面上的接触角。

选择 Define→Boundary Conditions 命令。

(7) 设置模拟具体的解参数。

选择 Solve→Controls→Solution 命令。

(8) 初始化解和为第二相设定初始体积份额。

选择 Solve→Initialize→Patch 命令。

(9) 计算求解和检查结果。

5.6.2 选用多相流模型并指定相数

为了选 VOF、混合，欧拉多相流模型，在 Multiphase Model(多相流模型)对话框(如图 5-3 所示)的 Model 选项组中选中 Volume of Fluid、Mixture 或者 Eulerian。

选择 Define →Models →Multiphase 命令，打开 Multiphase Model 对话框。

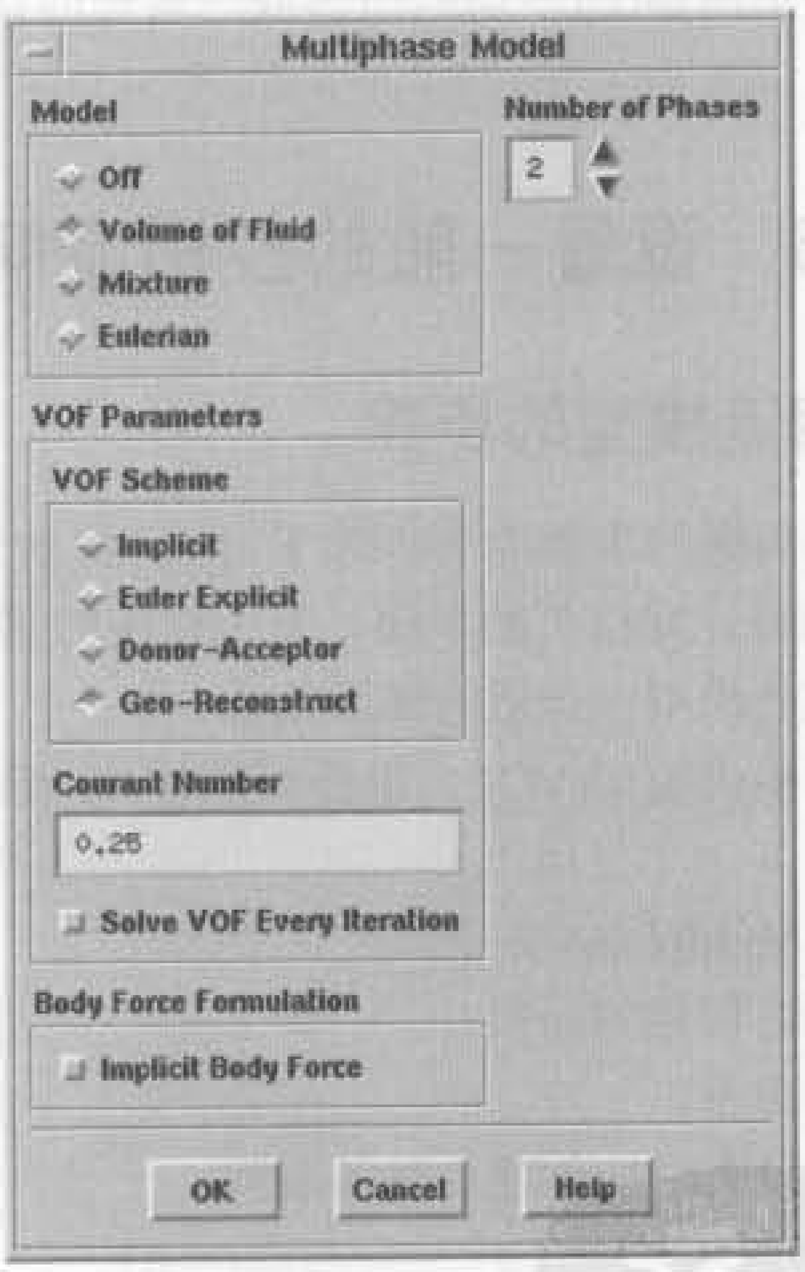


图 5-3 多相流模型对话框

这个对话框将扩展为显示已选择的多相流模型相关输入，对于不同的多相流模型，其展开是不同的。为了给多相流计算指定相数，在 Number of Phases 微调框中输入合适的值，最多可以指定 20 相。

5.6.3 选择 VOF 公式

为了指定使用的 VOF 公式，在多相流模型控制面板中 VOF Parameters 选项组中选择合适的 VOF Scheme。Fluent 中可用的 VOF 公式如下。

(1) Geo-Reconstruct：当求解结果对时间精确的瞬态行为感兴趣时，应当选择这个公式。Fluent 将自动为时间打开非稳态公式和一阶离散格式。

(2) Donor-Acceptor：如果网格中包含很多扭曲的六面体单元(hexahedral cells)，建议使用这个公式代替 Geo-Reconstruct 公式，能得到更精确的结果。应用这个公式时，Fluent 将自动为时间打开非稳态公式和一阶离散格式。

(3) Euler Explicit：由于 Donor-Acceptor 公式仅对四边形和六面体网格有效，它不能用于混合扭曲六面体单元。这种情况下，应使用 Euler Explicit 公式。这个公式也能用在 Geo-Reconstruct 公式不能给出满意结果的其他情形时，否则流动的计算变得不稳定。同时，Fluent 将自动为时间打开非稳态公式和一阶离散格式。

Euler Explicit 算法比 Geo-Reconstruct 算法有更少的计算耗费，相间的界面不再像用 Geo-Reconstruct 公式预测的那样明显。为减少这种扩散，建议将体积份额方程选为二阶离散格式。另外，为了获得更清晰的界面，在用 Euler Explicit 计算之后，可以考虑返回来用 Geo-Reconstruct 公式。

(4) 时间依赖的 Implicit 公式：如果要寻找稳态解和对中间的瞬态行为不感兴趣，但是最终的稳态解依赖于初始流动条件或者不让每相有明显的来流边界，可以选用这个公式。

选择 Implicit 公式，如果要考虑时间依赖，要在 Solver 控制面板中选择 Unsteady 求解。上面 Euler Explicit 算法的相关讨论也适用于时间依赖的 Implicit 公式。

(5) 稳态 Implicit 公式：如果要寻找稳态解和对中间的瞬态行为不感兴趣，并且最终的稳态解不受初始流动条件影响而每相有明显的来流边界，可以考虑选用这个公式。同样，上面对 Euler Explicit 公式的相关讨论也适用于稳态 Implicit 公式。

对于 Geo-Reconstruct 和 Donor-Acceptor 公式，如果使用了保角网格(也就是，在两个子边界相交的边界上网格节点的位置是一样的)，必须保证在这个区域内没有双边(0 厚度)壁面，如果有，必须分割它们。

5.6.4　定义均匀多相流

如果使用的是混合模型，可选择不做滑流速度的计算而求解一个均匀的多相流问题(也就是，所有相有着相同的速度)，默认情形，Fluent 会为第二相计算滑流速度。如果想求解一均匀多相流问题，在 Mixture Parameters 下关掉 Slip Velocity 选项。

5.6.5　包含气穴影响

对混合的欧拉模型计算，有可能包含气穴影响。为了选择气穴模型，在多相流模型控制面板中 Interphase Mass Transfer 下打开 Cavitation 选项。

由于气穴影响，应指定在使用传质计算时的两个参数，这些参数的指定应当与实际情况下的流动特征参数相一致：Reynolds Number 和 Cavitation Number。在多相流模型控制面板中 Cavitation Parameters 下面，设置 Vaporization Pressure(式(5-130)中的 p_v)和 Bubble Number Density(式(5-127)中的 η)。η 的默认值是 10000，由 Kubota 等推荐。默认的 p_v 值是 2367.8，为环境温度下水的汽化压力。

5.6.6 定义相概述

为了定义相(包括它们的材料属性)和相间的相互作用(例如，VOF 模型中的表面张力和壁面粘附，混合模型中的滑流速度函数，欧拉模型中的曳力函数)，将在相设置对话框中设置，如图 5-4 所示。

选择 Define →Phases 命令，打开 Phases 对话框。

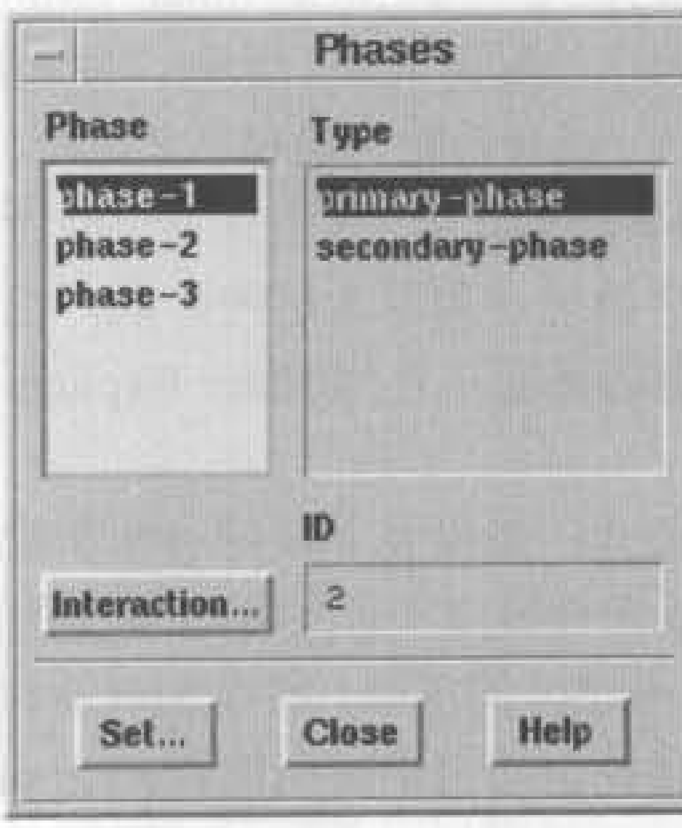


图 5-4 Phases(相设置)对话框

这个对话框中 Phase 列表框中显示每一相的名称；Type 列表框中显示相的类型：primary-phase 表明所选项是主相，secondary-phase 指出所选项是第二相。如果要指定相之间的相互作用，单击 Interaction 按钮设定。为 VOF 模型、混合模型、欧拉模型定义相及其相互作用的说明将在下面详细描述。

5.6.7 定义 VOF 模型中的相

通常，可以以任何方式指定主相和第二相，主要根据求解的具体问题而定。例如，对某区域多相中的一相，如果计划 patch 其初始体积份额为 1，指定这个相为第二相更方便。同样，如果一相是可压缩的，为了提高解的稳定性，建议指定它为主相。记住，只能有一相是可压缩的。

按以下步骤为 VOF 计算定义主相。

(1) 在 Phase 列表框中选择 phase-1。

(2) 单击 Set 按钮，进入 Primary Phase(主相设置)对话框，如图 5-5 所示。

(3) 在主相设置对话框中输入主相的名称。

(4) 在 Phase Material 下拉列表框中选择该相的材料。

(5) 定义或修改相的材料属性：单击 Edit 按钮，也可以在 Material 对话框中设置。

(6) 单击 OK 按钮关闭主相设置对话框。

按以下步骤为 VOF 计算指定第二相。

(1) 在 Phase 列表框中选择 phase-2 或除主相外的相。

(2) 单击 Set 按钮，进入 Secondary Phase(次相设置)对话框，如图 5-6 所示。

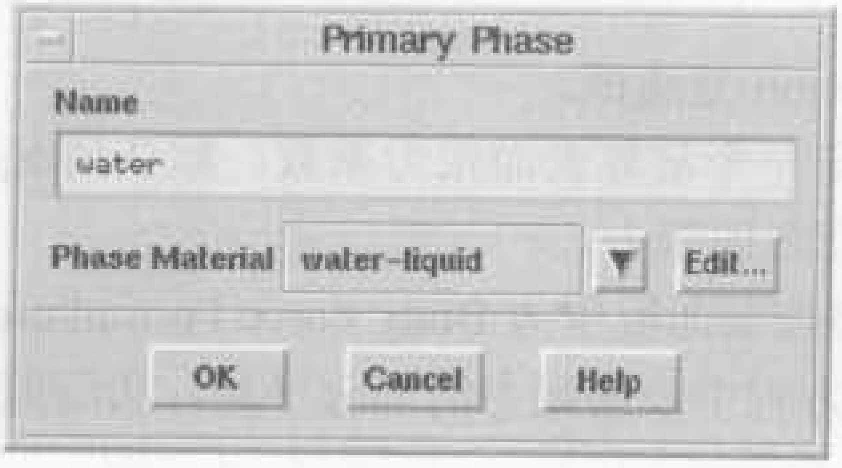


图 5-5　Primary Phase(主相设置)对话框

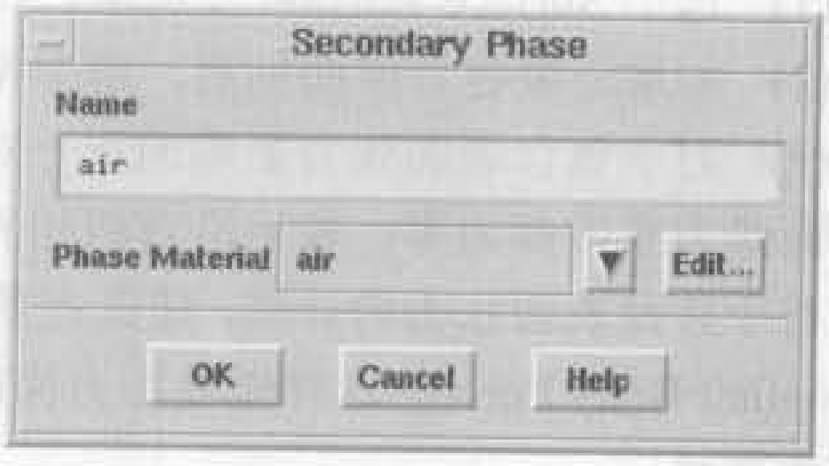


图 5-6　Secondary Phase(次相设置)对话框

(3) 在次相设置对话框中输入次相的名称。

(4) 在 Phase Material 下拉列表框中选择该相的材料。

(5) 定义或修改相的材料属性：单击 Edit 按钮，也可以在 Material 对话框中设置。

(6) 单击 OK 按钮关闭次相设置对话框。

关于包含表面张力和壁面粘附的影响，说明如下。

表面张力影响的重要性取决于毛细管数 *Ca*(见式(5-16))的值，或者韦伯数 *We*(见式(5-17))。如果 $Ca>>1$ 或者 $We<<1$，表面张力的影响可以忽略。

注意： 如果在表面张力有重大影响的计算区域内使用四边形或六边形网格，表面张力影响的计算会更精确。如果在整个区域内不能使用四边形或六边形网格，那么应当在影响区域内使用四边形或六边形的混合网格。

如果想包含沿着一对或多对相的界面上的表面张力的影响，单击 Interaction 按钮，打开 Phase Interaction(相间作用)对话框，如图 5-7 所示。设置沿着一对或多对相界面上表面张力的影响，遵循以下步骤。

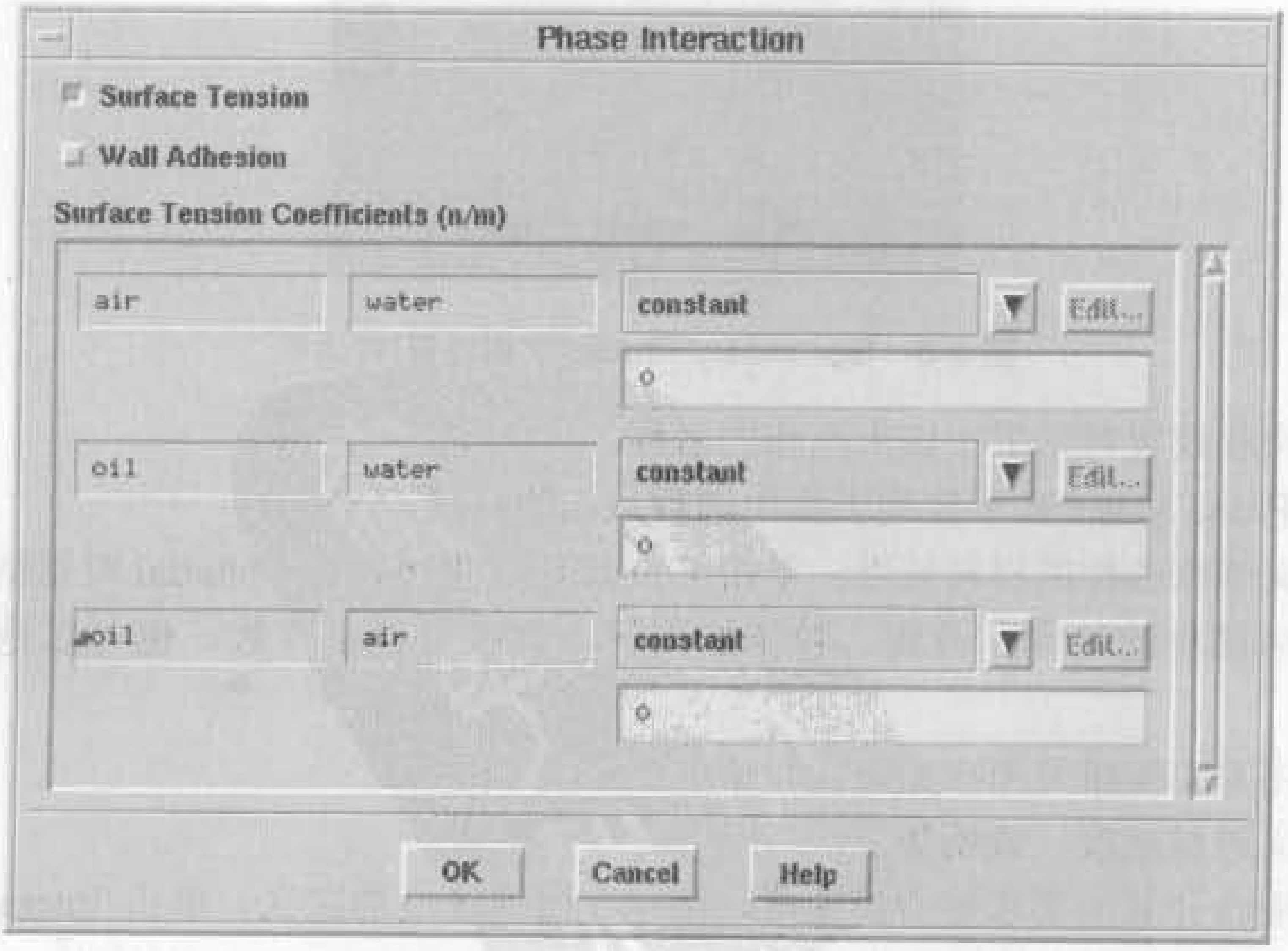


图 5-7　相间作用对话框

(1) 选中 Surface Tension 复选框。

(2) 如果还要包含壁面粘附，选中 Wall Adhesion 复选框。

(3) 对于包含表面张力影响的每一对相，指定一个常数表面张力系数。默认情况，所有表面张力系数都为零，表示沿着两相界面上没有表面张力。

对于涉及表面张力的计算，建议在多相流模型控制面板中为 Body Force Formulation 打开 Implicit Body Force。这样做使压力梯度和动量方程中表面张力的部分平衡，从而提高了解的收敛性。

5.6.8 定义混合模型中的相

在混合模型中也只能有一相是可压缩的。

在混合模型计算中定义主相的步骤与在 VOF 计算中相同。定义第二相的步骤有所区别，具体如下。

(1) 在 Phase 列表框中选择 phase-2 或除主相外的相。

(2) 单击 Set 按钮，进入 Secondary Phase(次相设置)对话框，如图 5-8 所示。

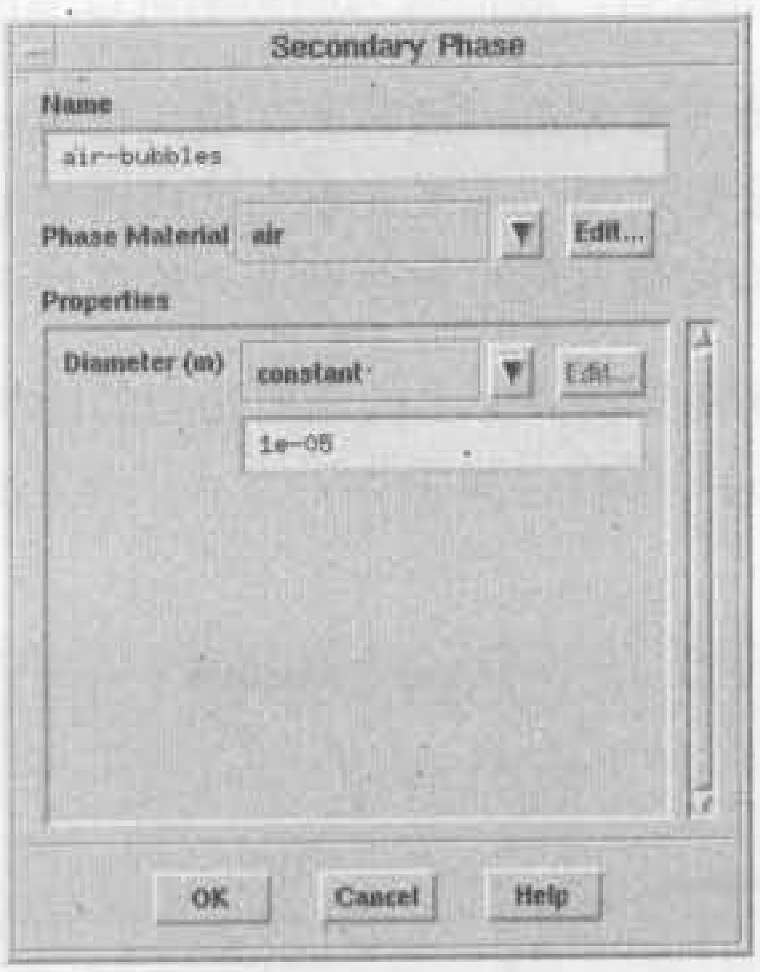


图 5-8 Secondary Phase(次相设置)对话框

(3) 在次相设置对话框中输入次相的名称。

(4) 在 Phase Material 下拉列表框中选择该相的材料。

(5) 定义或修改相的材料属性：单击 Edit 按钮，也可以在 Material 对话框中设置。

(6) 说明该相的气泡、液滴、粒子的直径，直径可以是常数，也可以是用户自定义函数。

(7) 单击 OK 按钮关闭次相设置对话框。

关于定义滑移速度，说明如下。

如果在混合计算中要求解滑流速度，需指定滑流速度的定义。单击 Interaction 按钮，打开如图 5-9 所示的混合模型的相间相互作用设置(Phase Interaction)对话框。

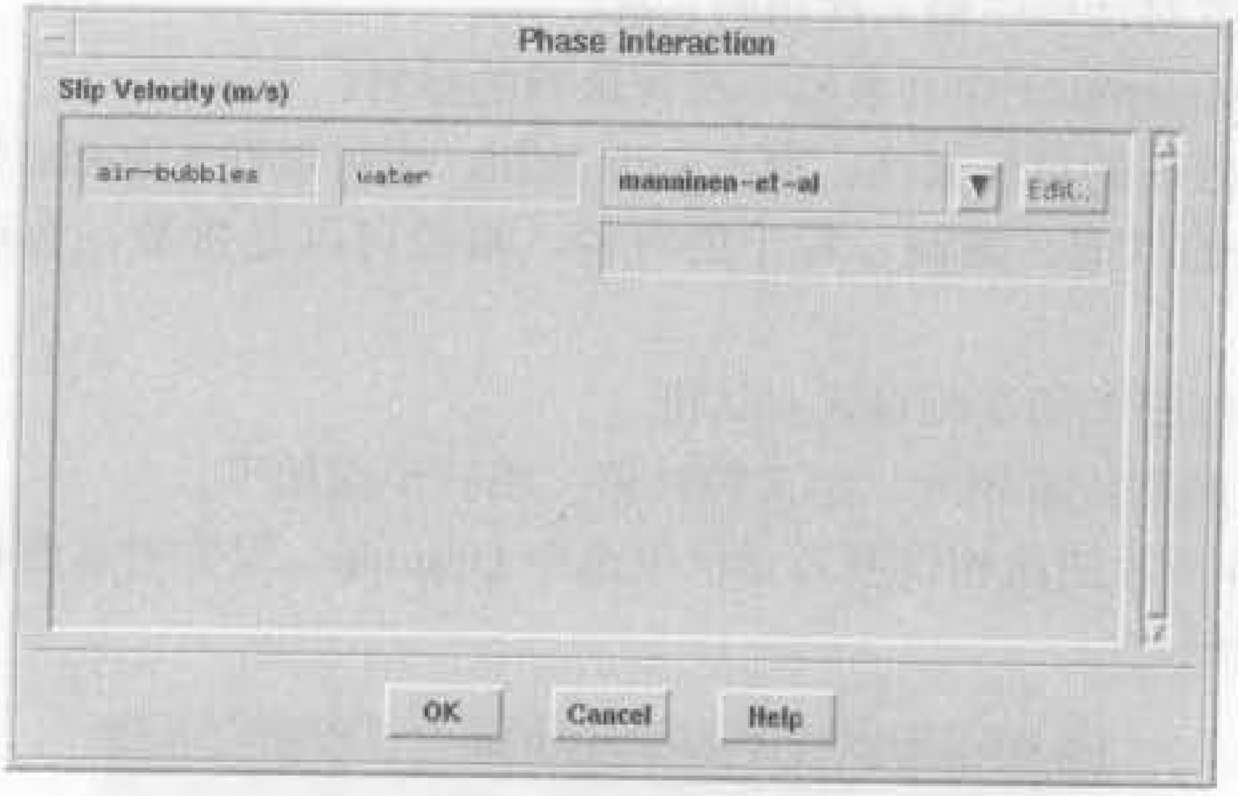


图 5-9　混合模型的相间相互作用设置对话框

在 Slip Velocity 下面，可以通过在附近下拉列表中选择合适的项目给与主相相关的第二相指定滑流速度函数。

(1) 选择 manninen-et-al(默认)，该算法是采用 Manninen 的代数滑移方法。

(2) 如果次相和主相有相同的速度，则选择 none 选项。

(3) 如果滑移速度是由自己定义，则选择 user-defined 选项。

5.6.9 定义欧拉模型中的相

在欧拉多相流计算中定义主相的步骤与在 VOF 计算中相同，定义次相与 VOF 模型有所差别。在欧拉多相流计算中定义非颗粒(即液体或气体)第二相的步骤如下。

(1) 在 Phase 列表框中选择 phase-2 或除主相外的相。

(2) 单击 Set 按钮，进入次相设置对话框，如图 5-10 所示。

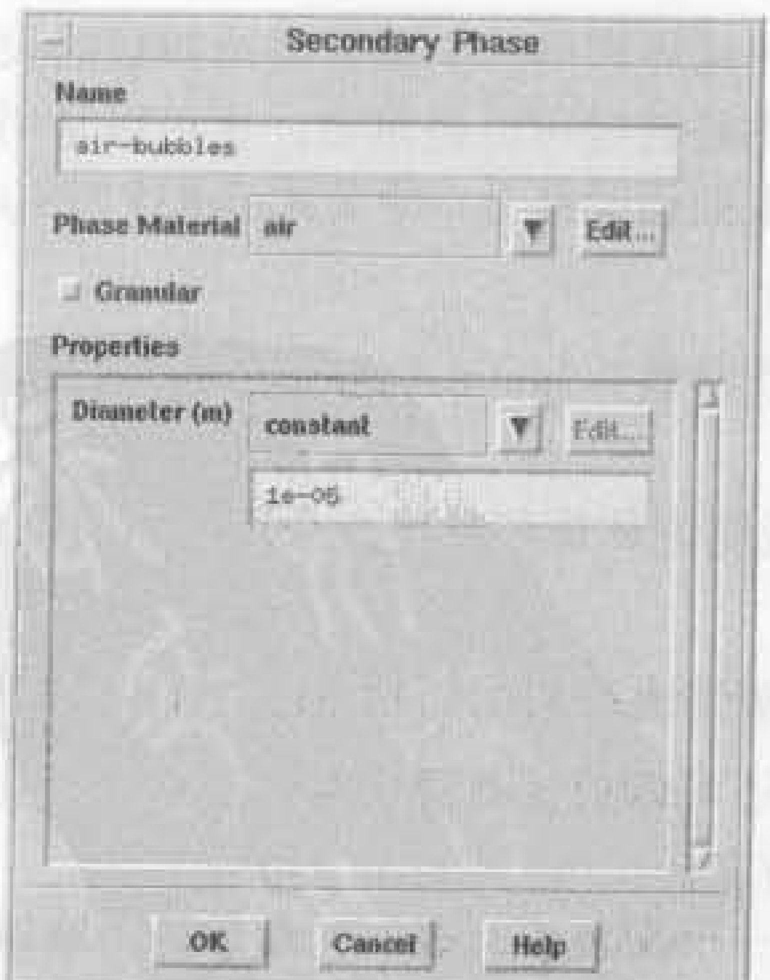


图 5-10　欧拉模型的次相设置对话框

(3) 在次相设置对话框中输入次相的名称。

(4) 在 Phase Material 下拉列表框中选择该相的材料。

(5) 定义或修改相的材料属性：单击 Edit 按钮，也可以在 Material 对话框中设置。

(6) 说明该相的气泡、液滴、粒子的直径，直径可以是常数，也可以是用户自定义函数。

(7) 单击 OK 按钮关闭次相设置对话框。

对于有颗粒相的欧拉多相流，定义颗粒第二相的步骤如下。

(1) 在图 5-10 所示的次相设置对话框中选中 Granular，设置对话框扩展成如图 5-11 所示的次相设置对话框。

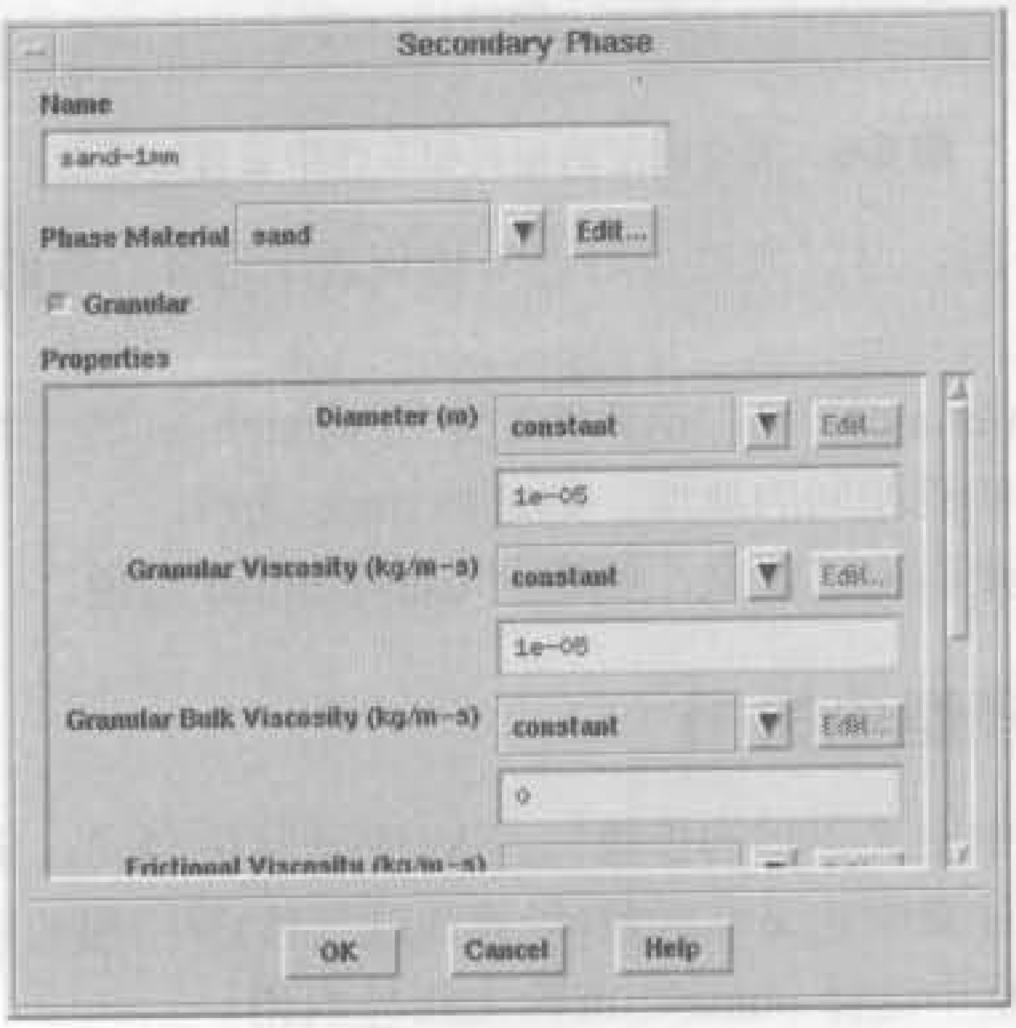


图 5-11　展开后的次相设置对话框

(2) 已有项的设置与前面描述相同，主要是介绍颗粒的设置情况。

- 指定颗粒的直径，颗粒直径可以是常数，也可以是用户采用 UDF 定义的。
- 指定颗粒的颗粒粘度的运动部分，即公式(5-83)中的 $\mu_{s,kin}$。可以选择常数，也可以采用 UDF 自定义的。
- 颗粒粘度，可选项有常数、Syamlal-obrien(式(5-85))、gidaspow(式(5-86))，或者是自定义。
- 颗粒体积粘度，即说明固体体积粘度(公式(5-41)中的 λ_q)，可选项有常数、lun-et-al，或自定义。
- 摩擦粘度，说明粘塑性流动的剪切粘度(公式(5-83)中的 $\mu_{s,fr}$)。如果选用 none，表明忽略摩擦粘度，还可以根据计算实际情况选用常数、Schaeffer(参见式(5-88))，或者采用 UDF 自定义。
- 内部摩擦角，给定一个常数值，表明角度 ϕ 是用摩擦粘度 Schaeffer 表达式(5-88)计算得到，这个选项只有在选用了 Schaeffer 摩擦粘度或者自定义粘度时出现。
- 充填限制，给出颗粒相的最大体积分数($\alpha_{s,\max}$)。对于单分散的球体，充填限制是

0.63(也是默认值)；对于多分散的情况，小球体可能充填在大球间的小间隙中，所以需要增大最大的充填限制。

定义相间的相互作用，对颗粒和非颗粒流动，必须指定在动量交换系数的计算中使用的曳力函数。对颗粒流，也必须指定颗粒碰撞的归还系数(restitution coefficients)。为颗粒和非颗粒流动包含可选的升力和虚拟质量力(下面描述)也是可能的。为指定这些参数，单击 Interaction 按钮，打开相间作用设置对话框，如图 5-12 所示。

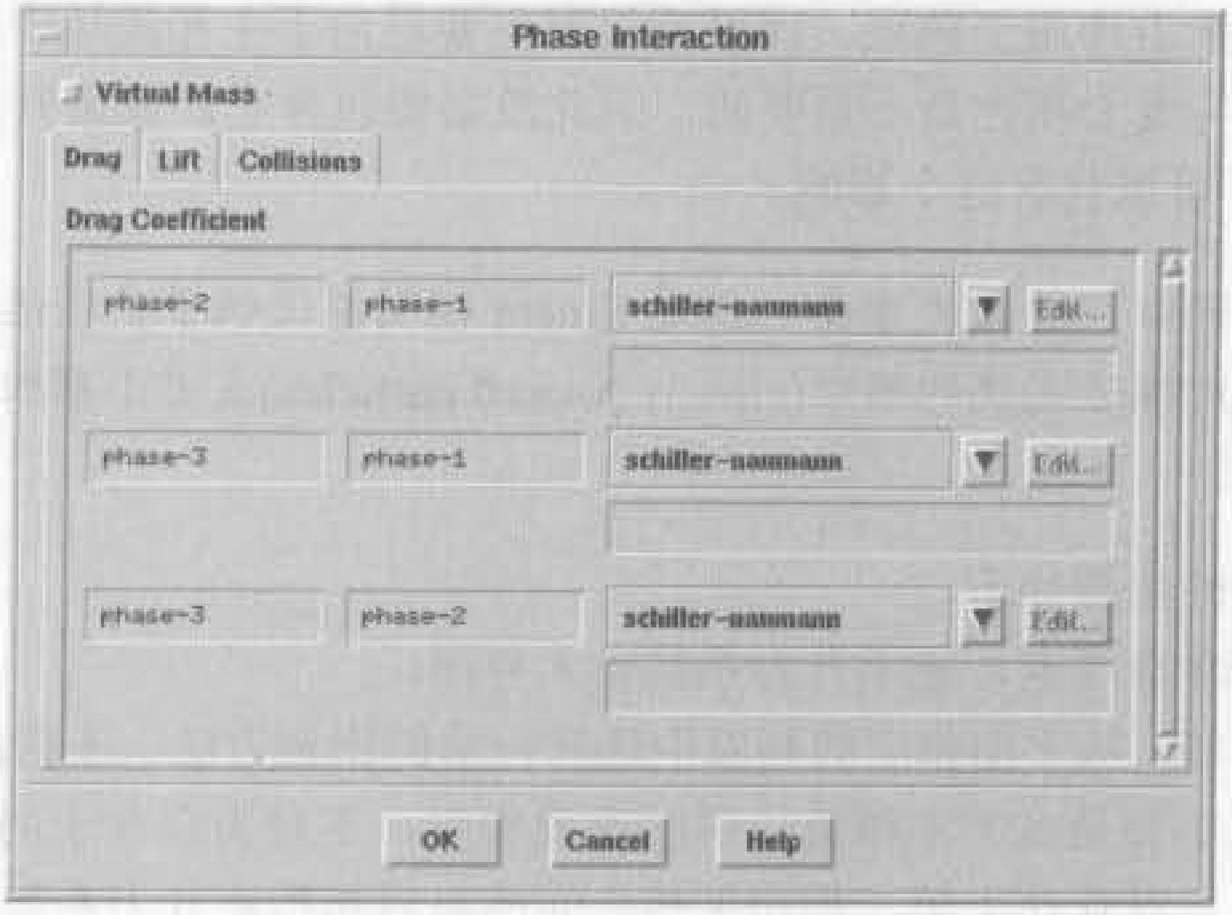


图 5-12　欧拉模型的相间相互作用设置对话框

Fluent 允许为一对相指定曳力函数，步骤如下。

(1) 切换到 Drag 选项卡，显示 Drag 函数的输入。

(2) 对每一对相，从下面相应的列表中选择合适的曳力函数。

- 对于公式(5-53)定义的液-液的曳力函数，选择 schiller-naumann 算法。这个算法是液-液多相流默认方法，也是通用算法。
- 对于公式(5-58)描述的液-液曳力函数，选择 morsi-alexander 算法。该算法最复杂，在大雷诺数范围内调整函数定义比较频繁，但这个模型的计算没有其他模型稳定。
- 选择公式(5-63)描述的液-液曳力函数，选择对称模型。这个模型适合于计算域的某个区域次相(分散相)变成计算域的另外一个区域主相的情况。
- 选择公式(5-73)描述的液-固曳力函数，选用 Wen and Yu 模型。该模型适用于总的次相体积分数明显低于主相的稀释相流动。
- 选择公式(5-76)描述的液-固曳力函数，选用推荐的浓缩流化床 gidaspow 的模型。
- 选择公式(5-66)描述的液-固曳力函数，选用 syamlal-obrien 模型。这个模型适用于与 syamlal-obrien 相关的粒子粘度。
- 选择公式(5-78)描述的固-固曳力函数，选用 syamlal-obrien-symmetric 模型。这个模型适用于两相均是固体的情况。
- 曳力函数是一个常数。
- 用户采用 UDF 定义曳力函数。
- 如果临时忽略相间的相互作用，选择 none。

说明恢复系数，对颗粒流，必须为颗粒间的碰撞指定恢复系数(式(5-78)中的 e_{ls} 和式(5-79)中的 e_{ss})。除了为每一对颗粒相之间的碰撞指定恢复系数外，也得为同相颗粒之间的碰撞指定恢复系数。其步骤如下。

(1) 单击 Collisions 条，显示恢复系数输入界面。

(2) 对于每一对相，说明常数的恢复系数，所有的恢复系数默认是 0.9。

对颗粒和非颗粒流，在第二相颗粒、液滴或气泡中包含升力(公式(5-43)中的 F_{lift})的影响是可能的。这些升力作用于颗粒、液滴或气泡主要是由于主相流场中的速度梯度。在大多数情形下，升力与曳力相比是不重要的，因此没必要包含它。如果升力重要(也就是说，如果相很快分离)，可以包含这个影响。

注意：对大颗粒，升力更重要，但是 Fluent 模型假设粒子直径远小于粒子间距离。这样对接近充满的颗粒(closely packed particles)或者小颗粒，包含升力是不合适的。

包含升力影响的步骤如下。

(1) 切换到 Lift 选项卡，显示升力系数输入界面。

(2) 对每一对相，从下面相应的列表中选择合适的指定方法。注意，既然作用于颗粒、液滴或气泡的升力主要是由于主相流场中的速度梯度，不必为存在于两个第二相间的每对相指定升力系数；只对存在于第二相和主相之间的每对相指定升力系数。

- 选择 none(默认)，即忽略升力的影响。
- 选择常数说明升力是常数值，在文本框中输入具体值。
- 选择 user-defined，表示升力系数用 UDF 定义。

对颗粒和非颗粒流，当第二相相对于主相加速时包含存在的虚拟质量力(公式(5-44)中的 F_{vm})是可能的。当第二相的密度远小于主相的密度时虚拟质量的影响是重要的(也就是对瞬态泡状柱流(transient bubble column))。

包含虚拟质量力的影响，选中欧拉模型的相间相互作用设置对话框中的 Virtual Mass 项。虚拟质量力的影响被包含在所有第二相内，使它仅为颗粒相是不可能的。

5.6.10 包含体积力

在许多情况下，相的运动部分是由于重力的影响。为了包含这个体积力，应在操作条件面板下选择 Gravity，并且指定 Gravitational Acceleration。

选择 Define→Operating Conditions 命令。

对于 VOF 计算，应当在操作条件面板下选择 Specified Operating Density，并且在 Operating Density 下为最轻相设置密度(这种排除了水力静压的积累，提高了 round-off 动量平衡计算精度)。如果任何一相都是可压缩的，设置 Operating Density 为零。

对于涉及体积力的 VOF 和 mixture 计算，建议在 Multiphase Model panel 下为 Body Force Formulation 选择 Implicit Body Force，这种处理通过解决压力梯度和动量方程中体积力的部分平衡提高了解的收敛。

5.6.11 为 VOF 模型设置时间依赖参数

在 Fluent 中，如果使用依赖时间的 VOF 公式，体积分数的一个隐式解或者在每一个时间步或者在每一迭代次数上获得，主要取决于输入的模型。也可以控制时间步用于体积分数的计算。

计算一个依赖时间的 VOF 解，必须在 Solver 面板下选择 Unsteady。如果选择了 Geo-Reconstruct、Donor-Acceptor 或 Euler Explicit 格式，Fluent 将自动打开 first-order 非稳态公式，因此不必再访问 Solver 面板。

选择 Define→Models→Solver 命令。

在 Multiphase Model 面板中，对于时间依赖的计算有两种输入。

(1) 默认情形，Fluent 将在每一时间步上求解体积分数方程一次。这意味着出现在其他输运方程中的对流流量系数不必在每一迭代层次上完全更新，因为体积分数在两次迭代之间不改变。

如果想让 Fluent 在一个时间步的每一迭代步上求解一次体积分数方程，应在 VOF 参数面板中打开 Solve VOF Every Iteration。当 Fluent 每次迭代是求解这些方程时，其他输运方程的对流流量系数将会在基于每次迭代更新体积分数的基础上更新。

通常，当其他流动变量在每一时间步收敛时，如果预计接触面的位置将发生变化，应当选择 Solve VOF Every Iteration。例如，当采用大的时间步长并希望达到稳态解，这种情形就会发生。如果采用小的时间步长，在每一迭代上执行求解体积分数的额外的工作是不必要的，因此可以让这些选项关闭。在两种选择中这种更可靠，并且与第一种选择相比，其在每一时间步上需要更少的计算付出。

如果使用的是滑动网格(sliding meshes)，使用 Solve VOF Every Iteration 会得到精度更高的结果，但是得付出更多的计算代价。

(2) 当 Fluent 执行时间依赖的 VOF 计算，用于体积分数计算的时间步长不必和用于其他输运方程的时间步长相同，Fluent 将会自动地为 VOF 调整时间步长，基于你为靠近自由表面输入的允许的最大 Courant 数。Courant 数是一个无量纲数，它是时间步长与计算流体通过控制容积的时间的比值：

$$\frac{\Delta t}{\Delta x_{\text{cell}} / v_{\text{fluid}}}$$

在流体接触面附近的区域，Fluent 通过外出流量的和分开各个单元的控制容积，其结果时间代表了流体流出控制单元变为空所用的时间。这些时间中最小的作为流体单元通过控制容积的特性时间。如上所述，基于这个时间和输入的允许最大的 Courant 数，在使用 VOF 计算时时间步被计算出来。例如，如果最大允许的 Courant 数为 0.25(默认)，时间步长将会至多被选为任何靠近接触面的最小通过时间的四分之一。

注意： 当采用隐式的求解方案时，这些输入是不要求的。

5.6.12 为欧拉多相流计算选择紊流模型

如果使用欧拉模型求解湍流，必须在 5.3.2 节中描述的 3 种湍流模型中选择一种模型

(见图 5-13)。

具体设置步骤如下。

(1) 在 Model 选项组中选中 k-epsilon 或者 Reynolds Stress 单选按钮。

(2) 选择理想的 k-epsilon 模型或 RSM 多相流模型，及其相关参数。

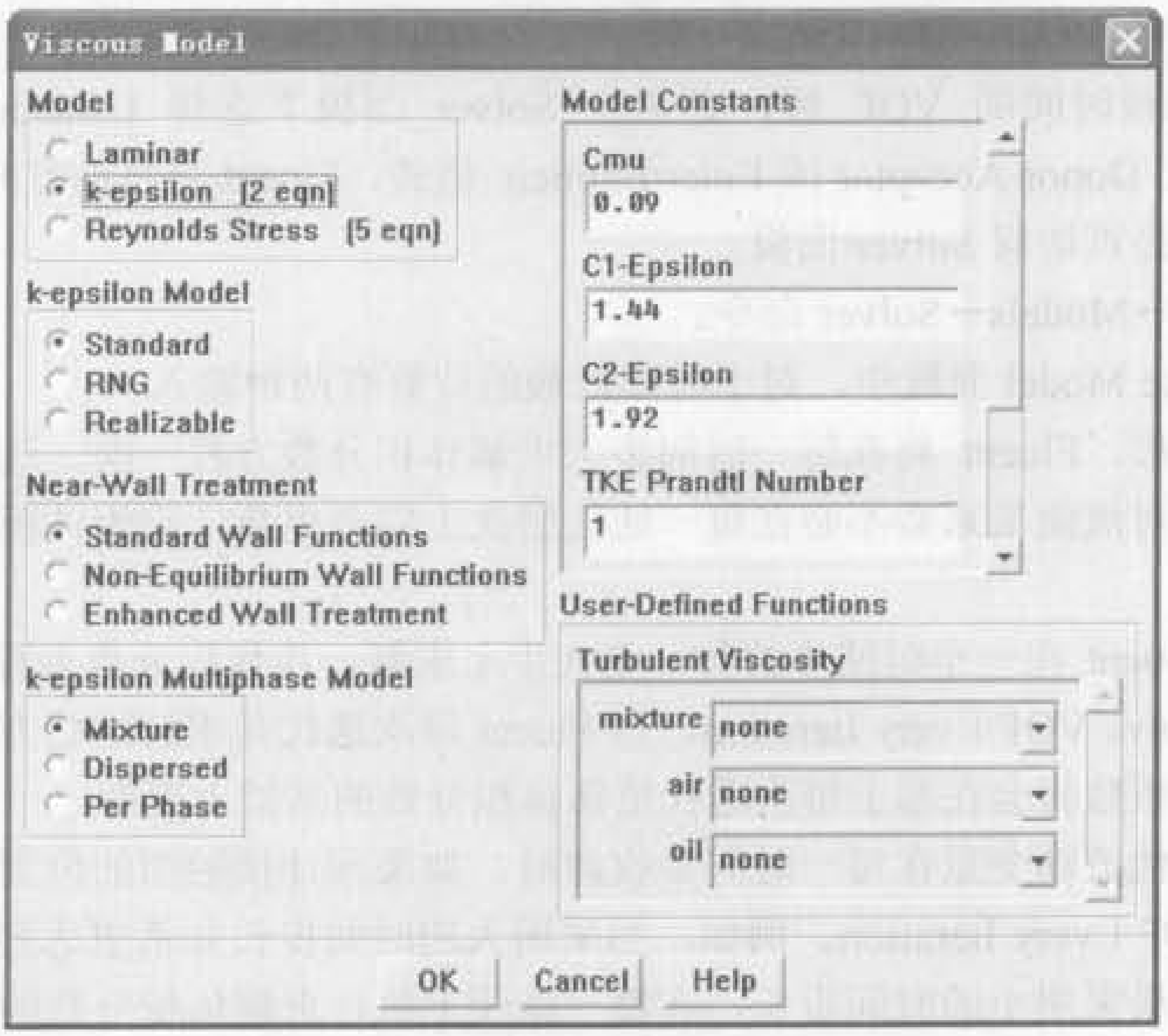


图 5-13　湍流模型设置对话框

(3) 在 k-epsilon 模型或 RSM 多相流模型下，说明理想的多相流湍流模型。

- 默认情况下，选择 Mixture 用于混合湍流模型。
- 选择 Dispersed 用于分散湍流模型，这个模型用于有一个明显主要的连续相，其余是分散稀释的次相。
- 也可以选择 Per Phase 用于 k-ε 湍流模型，这个模型适合于湍流输运在相中占主要地位。

默认情形，相间动量，κ、ε 源项不包含在计算中。如果想包含这些源项中的任一项，可以使用文本命令 define/models/viscous/ multiphase-turbulence/multiphase-options 进行设置。注意：包含这些项明显减慢收敛速度。如果要寻找额外的精度，应首先求得没有这些源项的解，接着包含上这些源项计算，大多数情形下这些源项可以忽略。

5.6.13　设置边界条件

多相流边界条件的设置在边界条件设置对话框(见图 5-14)中进行，但是设置多相流边界条件的步骤与单相流模型有些不同。必须分别为各个相设置一些条件，而其他的条件是所有相(也就是 mixture)所共享的，如下有详细的描述。

选择 Define→Boundary Conditions 命令

在 Fluent 计算中，需要分别为混合相(mixture)和各个单相指定边界条件，为混合相指

定的条件及为单相指定的条件取决于使用的三个多相流模型。每一个模型的详细叙述如下。

1. VOF 模型

如果使用的是 VOF 模型，必须为每一类型的区域指定的边界条件列举如下。

- 对于 exhaust-fan、inlet-vent、intake-fan、mass-flow-inlet、outlet-vent、pressure-outlet、velocity-inlet 这些边界，不必指定主相的条件。对于每一个的第二相，必须设置体积分数，可以为常数、型线分布或者用户定义函数(见 UDF 手册)，所有的其他条件都是为混合相指定的。

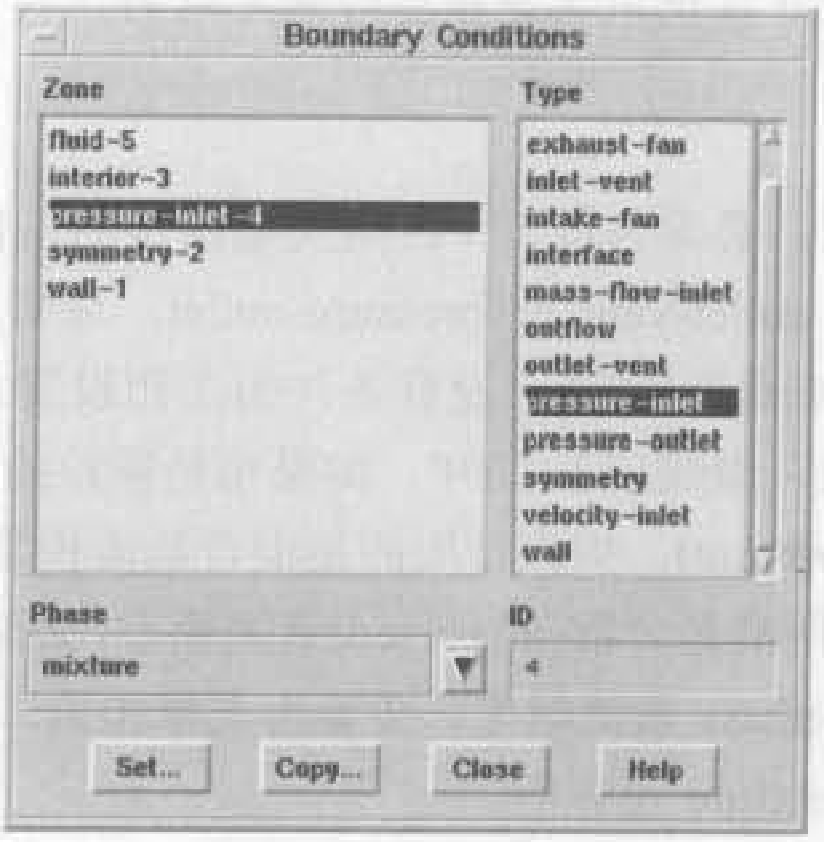


图 5-14　边界条件设置对话框

- 对于 axis、fan、outflow、periodic、porous jump、radiator、solid、symmetry、wall zone，所有的条件都是为混合相指定的。没有为单相指定的条件。
- 对于 fluid zone、mass sources，是为单相指定的，其他的所有 sources 都是为混合相指定的。如果 fluid zone 不是多孔的，所有其他条件都是为混合相指定的。如果 fluid zone 是多孔的，应在 Fluid 面板下为混合相选择 Porous Zone。Porosity inputs(如果相关)也是为混合相指定的。而 Resistance coefficients 和 direction vectors 分别为每一相指定。所有其他条件都是为混合相指定的。

2. 混合模型

如果使用的是混合模型，必须为每一类型的区域指定的边界条件列举如下。

- 对于 exhaust-fan、outlet-vent、pressure-outlet，不必为主相指定条件。对于每一个第二相，必须设置体积分数，可以设为常数、型线分布或者用户定义函数(见 UDF 手册)。所有的其他条件都是为混合相指定的。
- 对于 inlet-vent、intake-fan、mass-flow-inlet、pressure-inlet 这些边界，将为混合相指定描述其方向的方法(Normal to Boundary or Direction Vector)。如果选择 Direction Vector 描述方法，将指定坐标系统(仅适用 3D 情况)和每一相在流动方向的分量。对于每一个第二相，必须设置体积分数(如上面所述)。所有其他的条件都是为混合相指定的。

- 对于 axis、fan、outflow、periodic jump、radiator、solid、symmetry、wall zone，所有条件都是为混合相指定的，没有为单相指定的条件。
- 对于 fluid zone、mass sources，是为单相指定的，其他所有 sources 都是为混合相指定的。如果 fluid zone 不是多孔的，所有其他条件都为混合相指定；如果 fluid zone 是多孔的，将在 Fluid 面板下为混合相选择 Porous Zone。Porosity inputs(如果相关)也是为混合相指定的。而 Resistance coefficients 和 direction vectors 分别为每一相指定。所有其他条件都是为混合相指定的。

注意： pressure far-field 在使用混合模型时是无效的。

3. 欧拉模型

如果使用的是欧拉模型，必须为每一个区域类型指定的条件列举如下。

- 对于 exhaust-fan、outlet-vent 或 pressure-outlet，如果使用层流模型或使用混合紊流模型(默认的多相紊流模型)，没有条件为主相设置。对于每个第二相，必须设置体积分数为常数、型线或者 UDF。如果相是颗粒的(granular)，也必须设置颗粒温度(granular tempreture)。如果使用的是混合紊流模型，必须为 mixture 指定紊流边界条件；如果使用的是分散(dispersed)紊流模型，必须为主相设置紊流边界条件；如果使用的是 per-phase 紊流模型，必须为主相和第二相设置紊流边界条件。所有其他条件都是为 mixture 设置的。
- 对于 velocity-inlet，必须为每一相指定速度。对于第二相，必须设置体积分数。如果相是颗粒的(granular)，也必须设置颗粒温度(granular temperature)。如果使用的是 mixture 紊流模型，必须为 mixture 设置紊流边界条件；如果使用的是分散(dispersed)紊流模型，必须为主相指定紊流边界条件；如果使用的是 per-phase 紊流模型，必须为主相和第二相设置紊流边界条件。所有其他的条件都是为 mixture 设置的。
- 对于 axis、outflow、periodic、solid 或 symmetry zone，所有条件都是为 mixture 设置的，没有条件为单相设置。
- 对于 fluid zone, 所有 source terms 和 fixed values 都是为单相设置的，除非使用的是 mixture 紊流模型或 dispersed 紊流模型。如果使用的是 mixture 紊流模型，紊流的 source terms 和 fixed values 为 mixture 设置；如果使用的是 dispersed 紊流模型，只为主相设置。如果 fluid zone 不是多孔的，所有其他条件都是为 mixture 设置的。如果 fluid zone 是多孔的，需在 Fluid 对话框中为混合相选择 Porous Zone。Porosity inputs(if relevant)也是为混合相指定的。而 Resistance coefficients 和 direction vectors 分别为每一相指定。所有其他条件都是为混合相指定的。

注意： pressure far-field、fan、porous jump 和 radiator 边界在使用欧拉模型时是无效的。

设置边界条件的步骤如下。

(1) 在 Boundary Conditions 对话框的 Zone 列表框中选择边界。

(2) 如果必要，在这个边界上为 mixture 设置条件。

① 在 Phase 下拉列表框中选择 mixture。

② 如果当前该区域的类型正确，单击 Set 按钮，打开相应的对话框。如果当前该区域的类型不正确，在类型列表中选择正确的区域类型，并确认改变。

③ 在相应的对话框，说明混合相的边界条件。如图 5-15 所示为 Pressure Inlet 对话框。

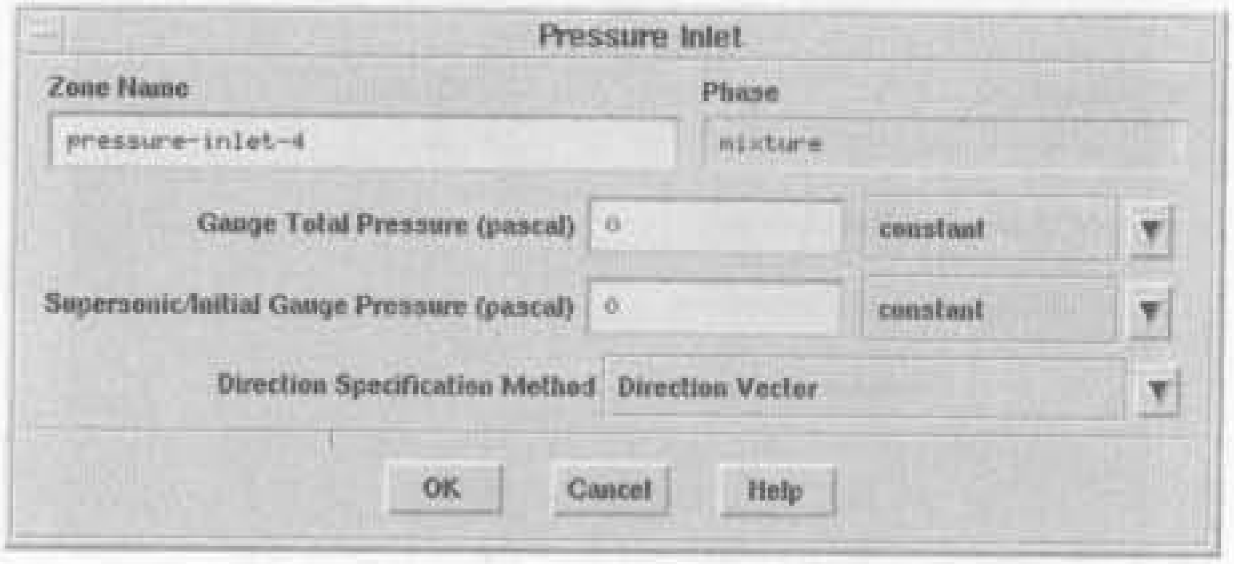


图 5-15　Pressure Inlet 对话框

对于 VOF 计算，如果在 Phase Interaction 对话框中选择了 Wall Adhesion，能在 wall 上把接触角指定到每一对相，如图 5-16 所示。

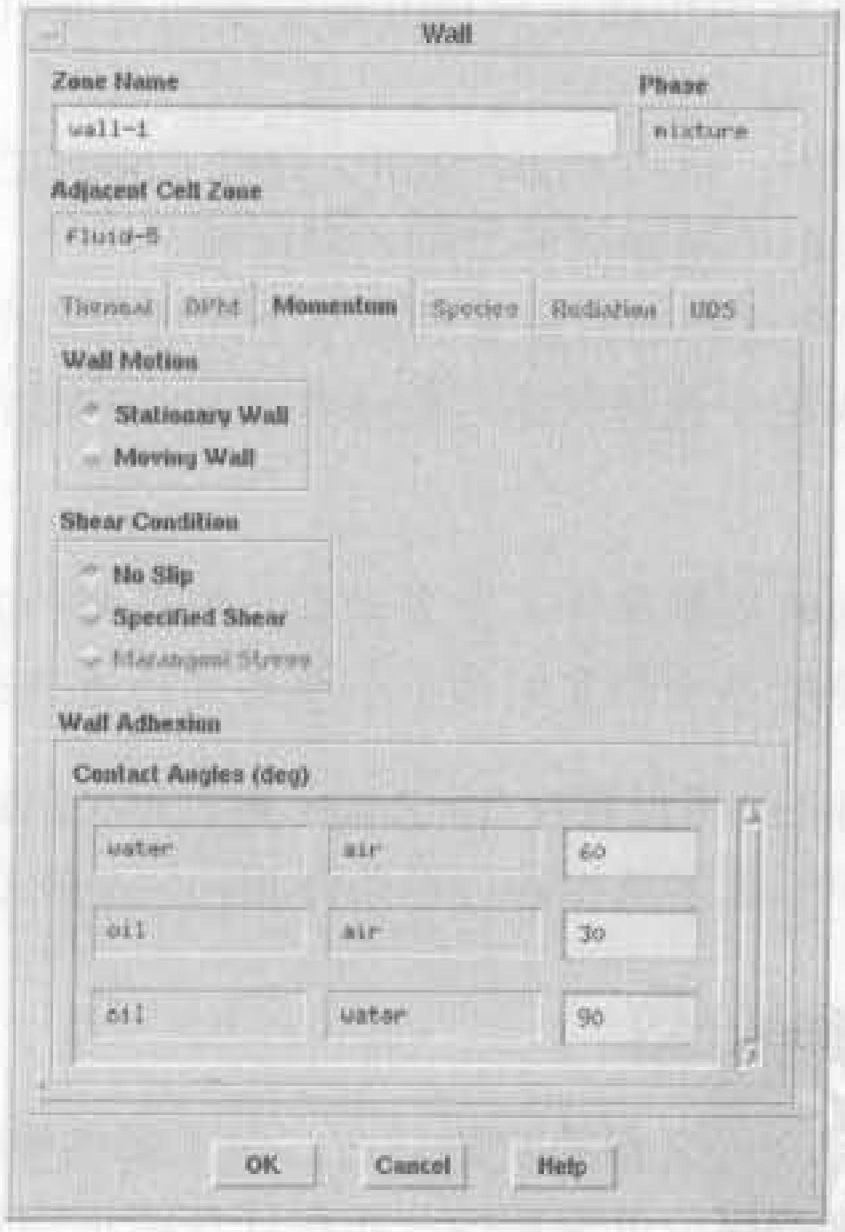


图 5-16　Wall 对话框

接触角(图 5-2 中的 θ_w)就是壁面和接触面切线的夹角，量度了在 Wall 对话框的成对的列表中第一相的值。例如，如果设置 oil 和 air 相的接触角在 Wall 对话框中(如图 5-16 所示)，θ_w 量度在 oil 相内。对于所有对默认值是 90°，就是没有壁面支持的影响(也就是，接触面垂直于支持面)。例如，接触角为 45°，相当于水沿着容器面爬行，通常是水在玻璃上。

④ 设置完混合边界条件，单击 OK 按钮。

⑤ 在 Phase 下拉列表框中选择相(如水)，当选择了单相中的一个(而不是 mixture)，只

有一类区域出现在 Type 列表中。在给定的边界上是不可能指定 phase-specific zone 类的，这种区域类型是为 mixture 指定的，它也适用于所有的单相。

⑥ 单击 Set 按钮，打开该相相关条件设置对话框，如图 5-17 所示为相的 Pressure Inlet 对话框。



图 5-17 Pressure Inlet 对话框

⑦ 说明该相的条件，注意这些条件仅适用该相。

⑧ 设置完该相条件后，单击 OK 按钮。

为多相流动复制边界条件的步骤与单相流动复制边界条件时有些不同。修改的步骤如下。

(1) 在边界条件对话框中，单击 Copy 按钮，打开 Copy BCs 面板。

(2) 在 From Zone 列表框中选择需要复制的源区域。

(3) 在 To Zone 列表框中选择需要被复制的单个区域或多个区域。

(4) 在 Phase 下拉列表框中选择所需复制条件的相，对于混合相或单个相都采用相同的操作。

注意： 对于一相复制的边界条件不会自动成为被复制的其他相和混合相的边界条件，如果需要对每一相复制每一个感兴趣的边界条件。

(5) 单击 OK 按钮。Fluent 将设置在 To Zones 列表框中所选区域所有选择相的边界条件与在 From Zone 列表框中所选区域相的边界条件相同。不能复制次设置条件，如热条件。

5.6.14 设置初始体积分数

一旦初始化了流动，就能定义相的初始分布。对于瞬态模拟，这个分布将作为在 t=0 时刻的初始条件；对于稳态模拟，设置初始分布在计算的早期阶段能提供更多的稳定性。

可以使用 Patch 面板为第二相充填(patch)初始体积分数。

选择 Solve→Initialize→Patch 命令。

如果想充填体积分数的区域已经作为隔离的单元区被定义，就只能充填那个地方的值。否则，可以创建一个包括合适单元的“寄存器”并在这个寄存器中充填值。

5.6.15 可压缩 VOF 和混合模型计算的输入

如果使用的是 VOF 或混合模型为可压缩流动，应注意如下几点。

- 只有相中的一个是可压缩的(也就是只有一相材料的密度可以选用理想气体定律)。
- 如果使用的是 VOF 模型，由于稳定性的原因，如果主相是可压缩的会更好(尽管是不需要的)。
- 如果在边界上指定总压力(也就是 pressure inlet or intake fan)，在那个边界上的具体的温度值对于可压缩相将会使用 total temperature，而对其他相使用 static temperature(它们是不可压缩的)。

5.6.16　凝固/熔解 VOF 计算的输入

如果在 VOF 计算中包含了熔解和凝固，应注意如下两点。

- 在单相和多相下模拟熔解和凝固是可能的。
- 对没有熔解或凝固的相，必须设置潜热 L(latent heat)、液相温度(T_{liquidus})和固相温度(T_{solidus})为零。

5.7　一般多相流问题的求解策略

5.7.1　VOF 模型的求解策略

为了提高 VOF 模型求解的精度和收敛性，列举如下几条建议。

1. 设置参考压力位置

参考压力的位置应该移动到能减少压力计算的位置。默认的情况，参考压力的位置在单元中心或靠近点(0,0,0)。可以通过指定新的 Reference Pressure Location 在 Operating Conditions panel 中移动这个位置。

选择 Define→Operating Conditions 命令。

选择的位置应当在这样的区域，那里总是包含密度最小的流体(如果计算的有一个气相和一个或多个液相也就是气相)。这是因为当给定相同的速度分布时高密度流体的静压变化大于低密度流体。如果相对压力为零的区域出现在压力变化小的区域，将比压力变化出现在大的非零值的区域带来少的计算量。例如，在包含空气和水的系统，参考压力的位置选在充满空气的区域而不选在充满水的区域是非常重要的。

2. 压力插值方案

对所有的 VOF 计算，应当使用 body-force-weighted pressure interpolation scheme or the PRESTO! Scheme。

选择 Solve→Controls→Solution 命令。

3. 为隐式和欧拉显式方程选择离散方案

当使用 the implicit 或者 Euler explicit scheme 时，为了提高相间界面的清晰度(sharpness)，应当采用 second-order 或 QUICK discretization scheme 离散体积分数方程。

选择 Solve→Controls→Solution 命令。

4. 对于瞬态方程，采用 Pressure-Velocity Coupling 和 Under-Relaxation

设置求解器的另一个变化是在使用的速度压力耦合方案和欠松弛因子中。通常瞬变流计算建议采用 PISO 方案。使用 PISO 时允许增加所有欠松弛因子的值，而不会减弱解的稳定。通常能增加所有变量的欠松弛因子到 1 并且能达到预期的稳定和收敛速度(要求每一时间步内用较少的迭代次数)。对于在四边形和三角形网格上的计算，用 PISO 方案时为了提高稳定性建议为压力选择欠松弛因子为 0.7～0.8。

选择 Solve→Controls→Solution 命令。

当用 Fluent 进行任何模拟时，如果欠松弛因子设置为 1 时，解出现不稳定、发散行为，欠松弛因子必须减小。提高稳定性的另一个方法是减小时间步长。

5. 设置稳态方程的松弛因子

如果使用稳态隐式的 VOF 方案，为了提高稳定性，所有变量的欠松弛因子应设置在 0.2～0.5。

5.7.2 混合模型的求解策略

1. 设置滑移速度的松弛因子

应当为滑流速度选用较低的欠松弛因子开始混合模型的计算。建议采用 0.2 或更小。如果解显示出好的收敛行为，可逐渐增加这个值，直到合理。

2. 计算初始条件

对某些情况(如旋风分离)，如果开始计算时不求解体积分数和滑流速度方程，会更快地获得解。一旦启动了混合模型，能暂时使这些方程无效而开始初始计算。

选择 Solve→Controls→Solution 命令。

在 Solution Controls 对话框下，在 Equations 选项组中不选中 Volume Fraction 和 Slip Velocity，这样开始计算初始流场。一旦收敛的流场获得了，再打开 Volume Fraction 和 Slip Velocity 方程，开始混合计算。

5.7.3 欧拉模型的求解策略

为了提高收敛性，在求解完整欧拉多相流模型前可以先获得初始解。有 3 方法可以用来为欧拉多相流计算获得初始解。

(1) 启动和求解问题用混合模型(选或不选滑流速度都可)代替欧拉模型。然后启动欧拉模型，完成设置，采用混合模型的解作为起点继续计算。

(2) 通常启动欧拉多相流计算，但是仅计算主相的流动。这样做时，在 Solution Controls 面板的 Equations 下面不选体积分数，一旦为主相获得了初始解，打开体积分数方程继续为各相计算。

注意：没有获得用混合模型或欧拉模型作为欧拉多相流模型的初始解，不应该使用

单相解。这样做，不能提高收敛性，可能会给流动的收敛带来更多的困难。

(3) 如果计划在稳态欧拉多相流模拟中包含升力和虚拟质量力，经常减弱问题的稳定性，这有时发生在计算的早期阶段，是由于暂时忽略了升力和虚拟质量力引起的。一旦没有这些力，解开始收敛。因此，可以打断计算，合适地定义这些力，继续计算。

第 6 章　多相流计算实例

6.1　沉淀池活性污泥沉降的计算

6.1.1　问题描述

二次沉淀池是污水处理过程中重要的组成部分，其中的流动情况以及悬浮活性污泥在池内的浓度分布受沉淀池的结构和入口水流流速的影响，而通过实验研究获得这些参数有一定困难，数值模拟技术随着计算技术的发展逐渐成熟，能为沉淀池设计中的不确定因素确定提供依据。

沉淀池的数值模拟起源于 20 世纪 80 年代初，Schamber、Larock 和 Imam 等，Adams 和 Rodi 等应用湍流模型对初沉池的速度场进行模拟计算及实验研究；我国的曾光明等利用涡量-流函数法建立二次沉淀池的控制方程，以控制容积法对方程进行离散，求出了其中的速度分布场及浓度分布场；蔡金傍等采用有限元方法对平流式沉淀池进行了数值模拟分析。但实质上，二次沉淀池中有污水和悬浮活性污泥，其流动为液固两相流，上述数值计算分析仅局限于流动和近似的浓度计算，无法真实模拟池中的两相流动情况。屈强等采用液固两相流模拟了辐流式二沉池的污泥和水的流动，本书也采用液固两相流来计算辐流式二沉池的活性污泥的沉降情况。

某辐流式二次沉淀池中心深 4.0m，周边深 2.0m，泥斗设在池中央，池底向中心倾斜，一块垂直挡板，用于把来流引向池底，一块水平挡板，用于防止进水和污泥回流之间的短流。本模型属轴对称图形，几何尺寸如图 6-1 所示。

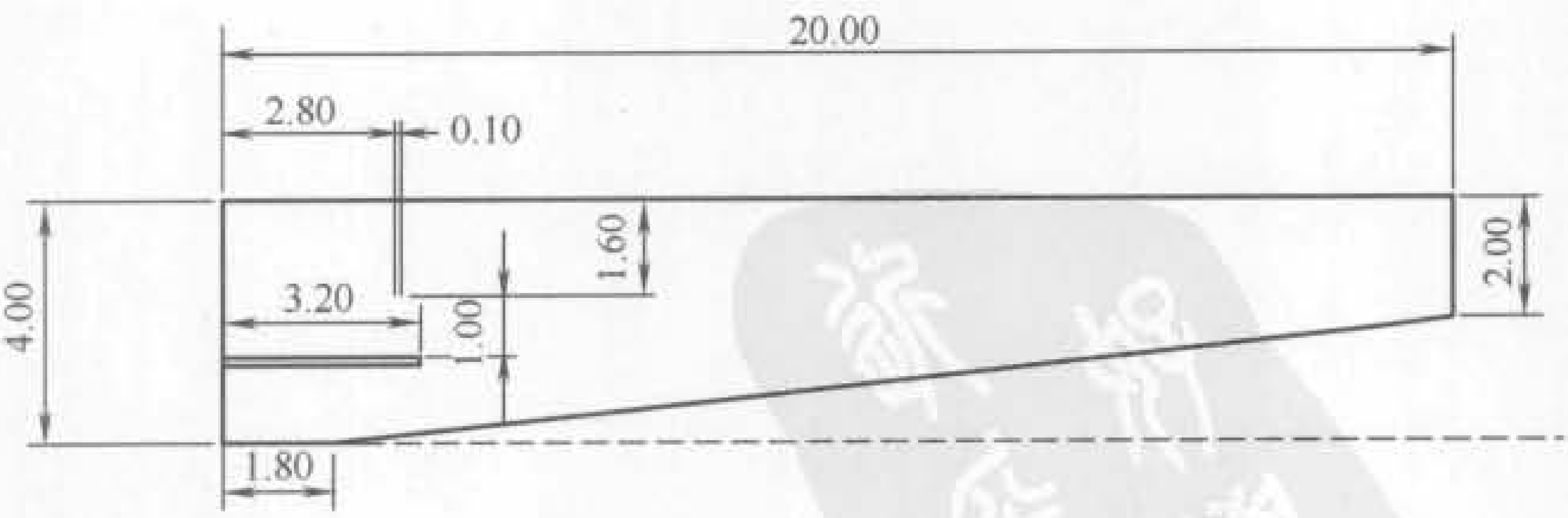


图 6-1　计算区域截面尺寸

进水的密度为 1000.35kg/m^3，动力粘度为 1.005e^{-3} Pa·s；污水的密度为 1051kg/m^3，动力粘度为 2.001e^{-2} Pa·s 。

6.1.2　具体计算

1. 建模

在 Pro/ENGINEER 中建模，如图 6-2 所示，输出为 mf_1.sat 文件。

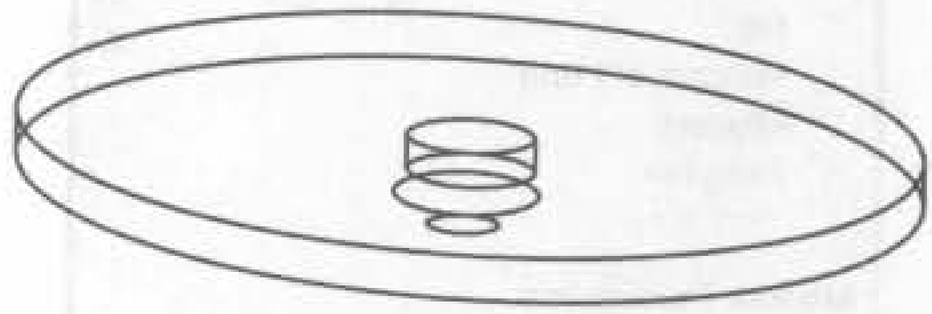


图 6-2　计算实体模型

2. 网格划分

将上述 mf_1.sat 文件 import(导入)到 Gambit 中，复制顶面圆的两个点，Y 方向向下位移 0.4，另外创建一点(0,3.6,20)，根据这三点做一个圆，而后形成一个面，将实体用此面切开。

下面的体采用四面体网格进行划分，在 Mesh Volumes 对话框中，Elements 选择 Tet/Hybrid，Type 选择 TGrid，interval size 设置为 0.5，共划分出 31437 个节点，148919 个单元；上面的两个体采用六面体网格划分，在 Mesh Volumes 对话框中，Elements 选择 Hex/Wedge，Type 选择 Cooper，interval size 设置为 0.5，外部大些的实体共划分出 11398 个节点，11110 个单元，中间小些的实体共划分出 286 个节点，248 个单元。网格模型如图 6-3 所示。

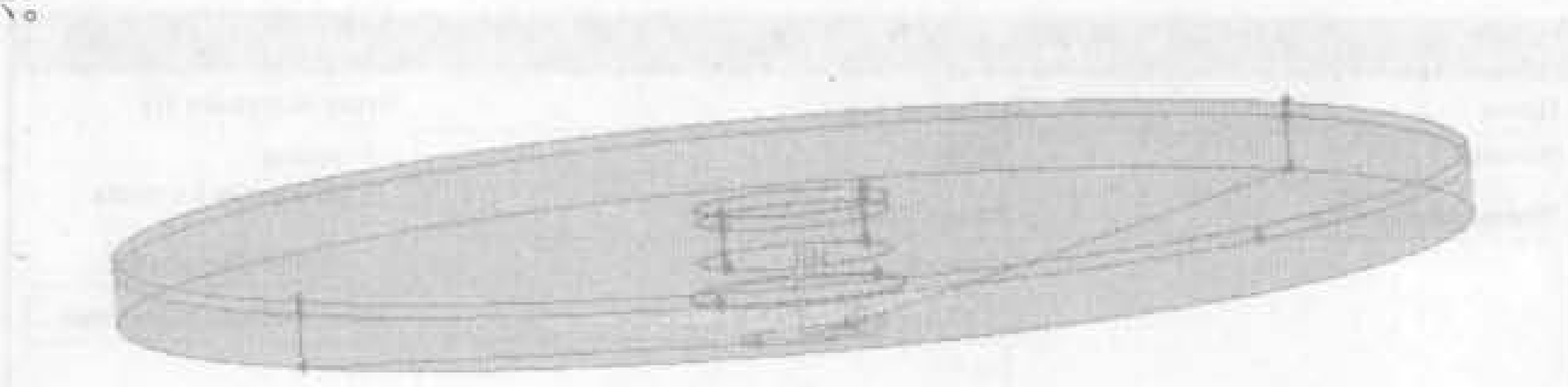


图 6-3　网格模型

在 Specify Boundary Types 对话框，顶面中心面设置为 VELOCITY_INLET 边界，命名为 waterin，侧面圆周的上部圆周面(即先前切开的部分)设置为 PRESSURE_OUTLET 边界，命名为 waterout，底部的圆面设置为 PRESSURE_OUTLET 边界，命名为 sedimentout，上面外部的大圆面设置为 WALL，命名为 freeface。保存后输出为 msh 文件，命名为 mf_1.msh。

3. Fluent 设置

1)　读入 msh 文件

选择 File→Read→Case 命令，读入 Gambit 输出的 mf_1.msh 文件，并检查网格(Grid→Check)，注意最小体积要大于零。选择 Grid→Smooth/Swap 命令，消除网格的扭曲。

2)　模型设置

选择 Define→Models→Solver 命令，打开 Solver 对话框，属于稳态计算，保持默认

设置。

选择 Define→Models→Multiphase 命令，打开 Multiphase Model 对话框，选择 Mixture 多相流模型，Number of Phases 设置为 2，如图 6-4 所示。

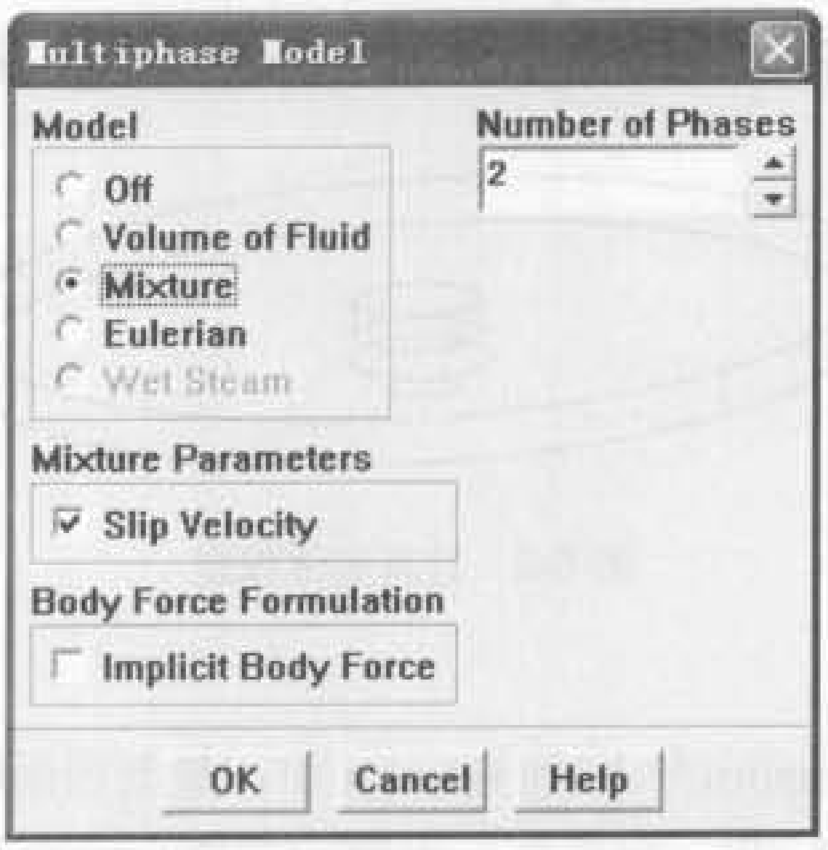
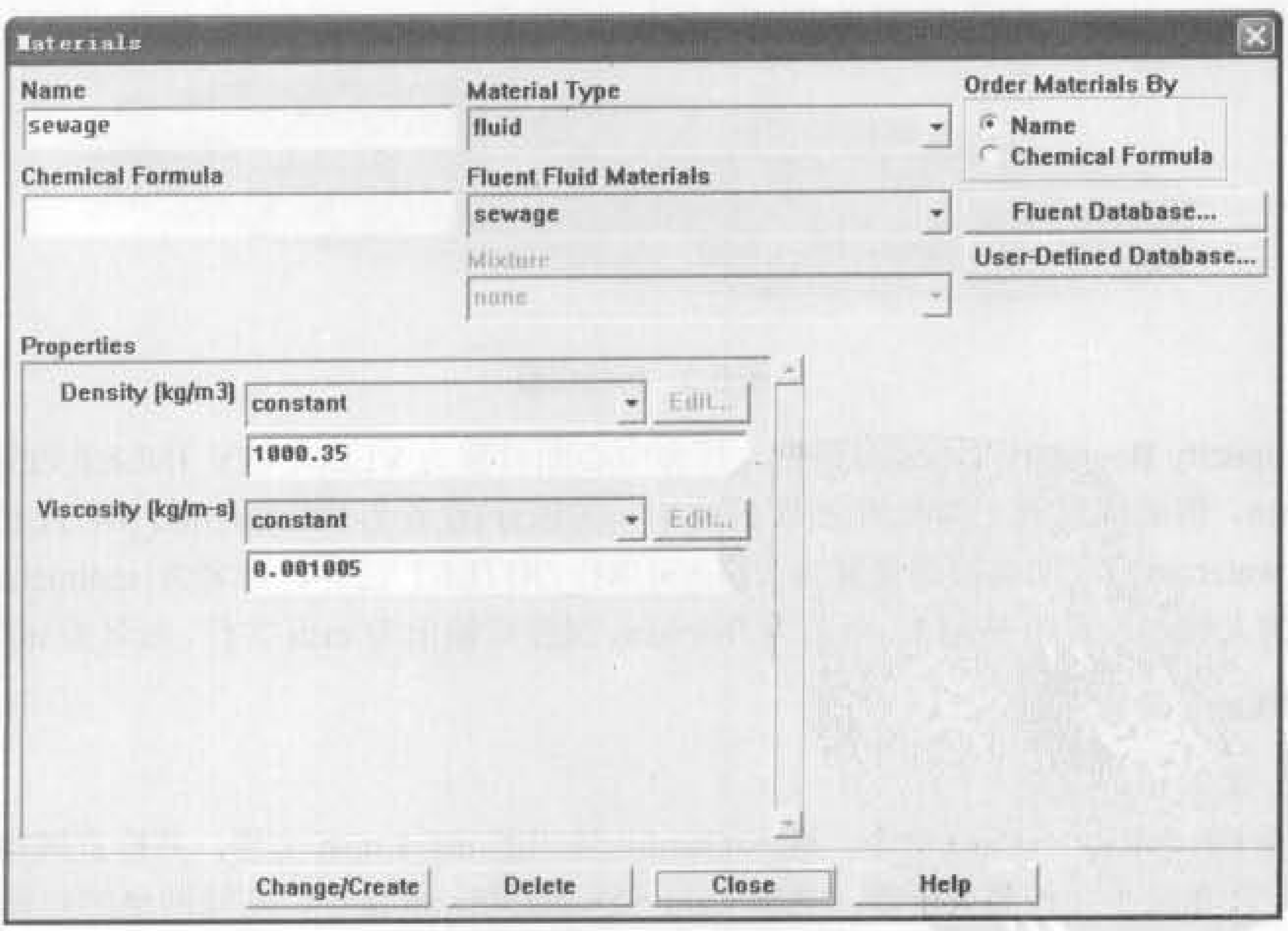


图 6-4 启动多相流模型

选择 Define→Models→Viscous 命令，打开 Viscous Model 对话框，选择 k-epsilon 模型，其余保持默认设置。

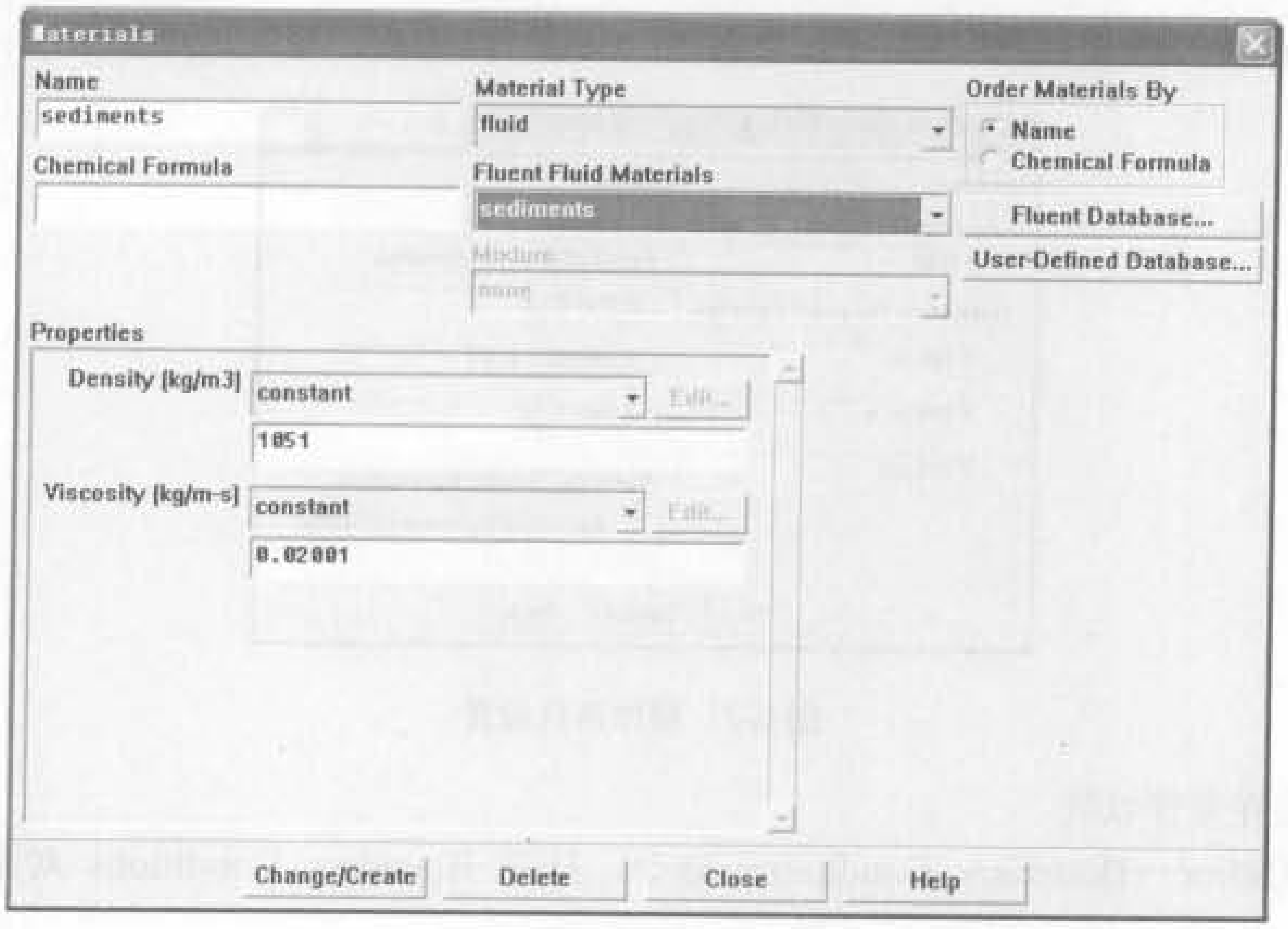
3) 材料设置

选择 Define→Materials 命令，打开 Materials 对话框，如图 6-5 所示，分别创建 sewage 和 sediments 材料。



(a)

图 6-5 第 1 相材料创建



(b)

图 6-5 (续)

4) 相设置

选择 Define→Phases 命令，打开 Phases 对话框，将 sewage 设置为主相(primary-phase)，将 sediments 设置为次相(secondary-phase)。选中 Granular 复选框，在 Diameter 栏的文本框中输入 0.001，在 Granular Viscosity 栏的文本框中输入 0.02001，在 Packing Limit 栏的文本框中输入 0.5，如图 6-6 所示。

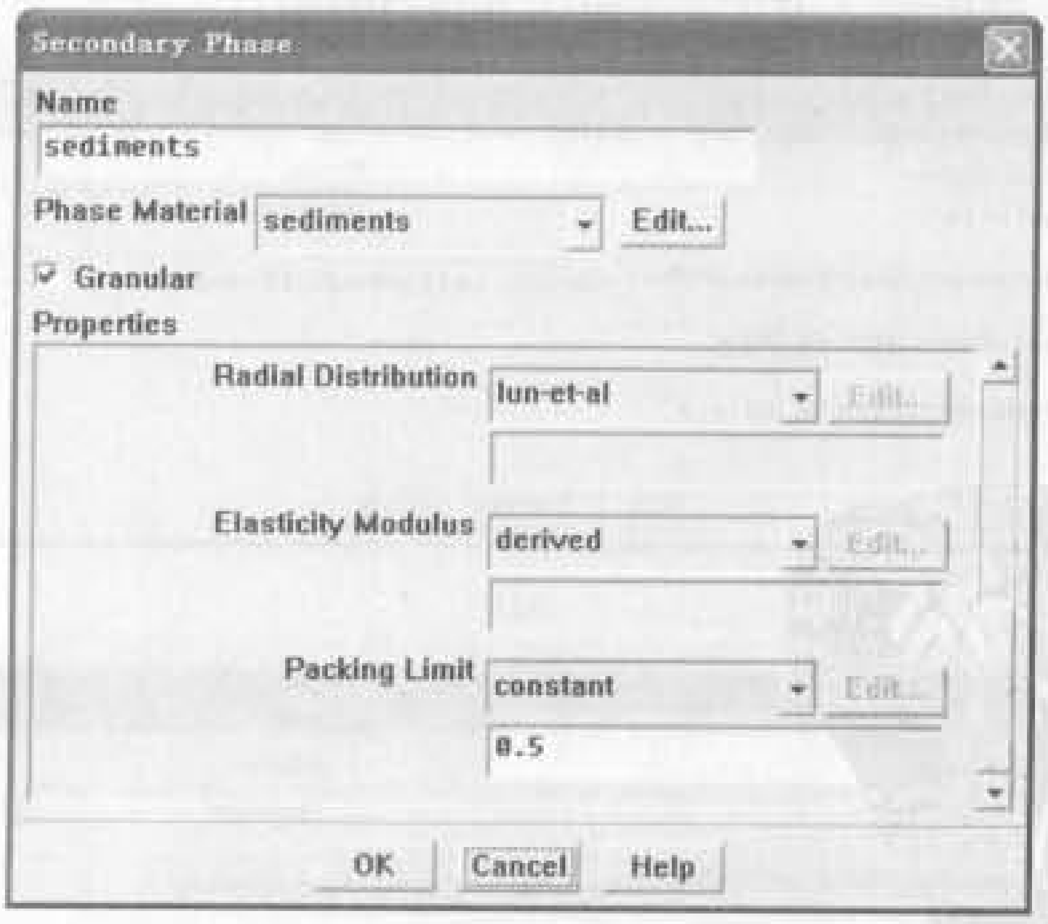


图 6-6 第 2 相材料创建

5) 操作条件设置

选择 Define→Operating Conditions 命令，打开 Operating Conditions 对话框，设置参考压力的位置为 X=0，Y=3.9，Z=20，选中 Gravity 复选框，设置 Y 方向的重力加速度为-9.81，

如图 6-7 所示。

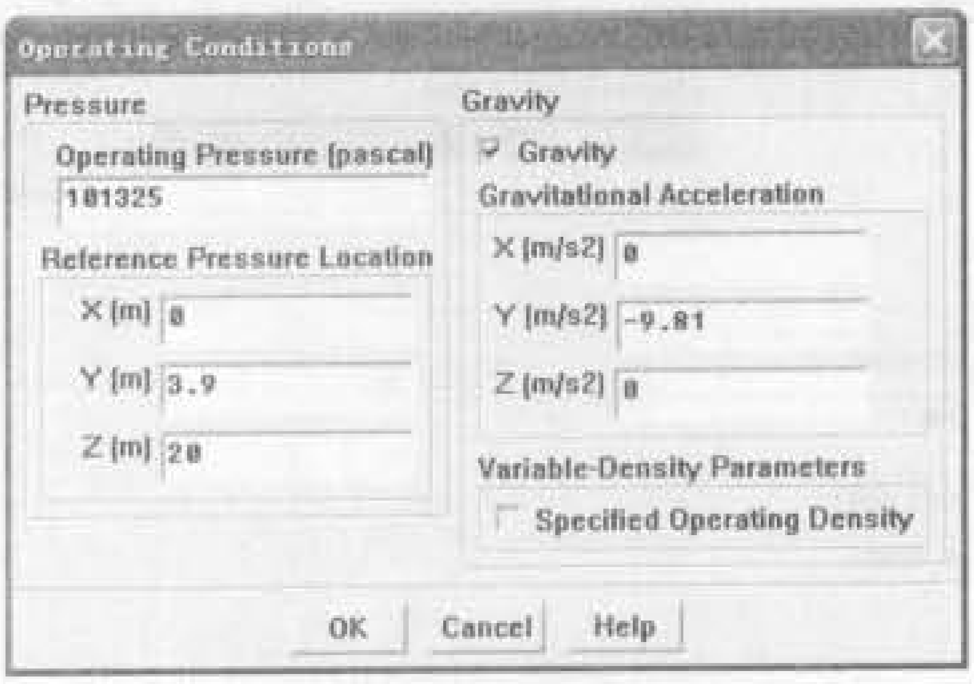
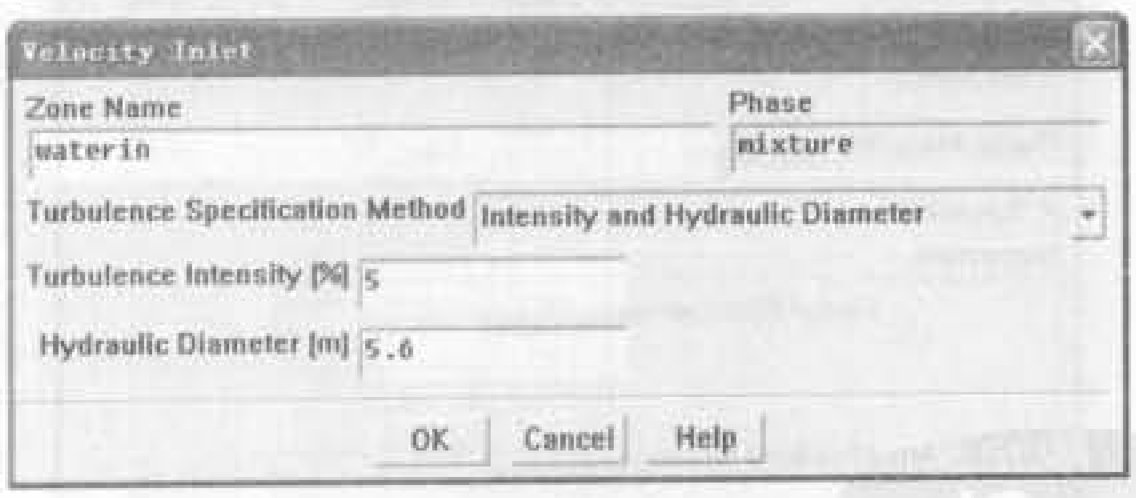


图 6-7 操作条件设置

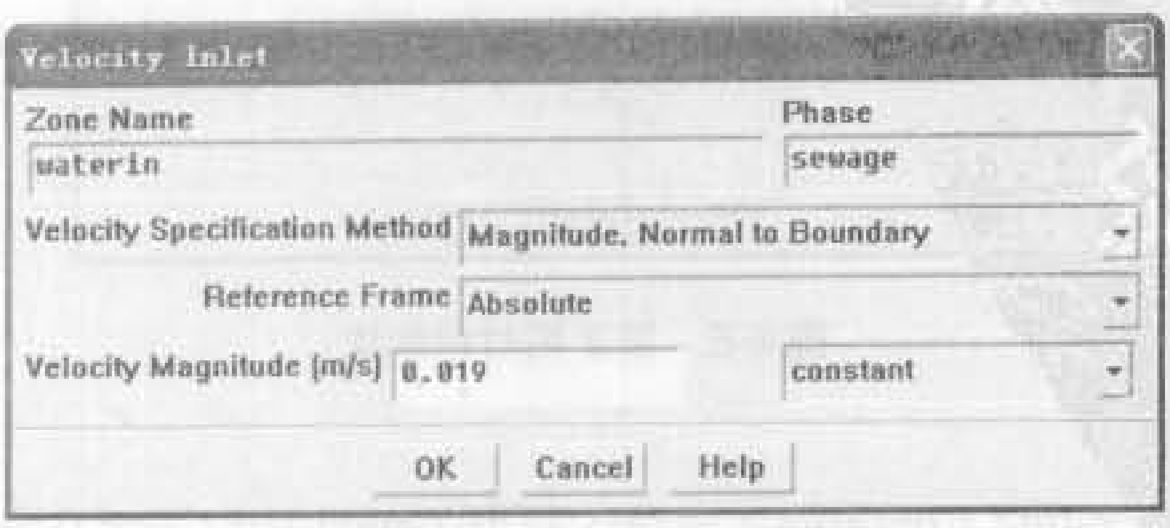
6) 边界条件设置

选择 Define→Boundary Conditions 命令，打开 Boundary Conditions 对话框(可参阅图 2-18)。

在 Zone 列表框中选择 waterin，在 Phase 下拉列表框中选择 mixture，单击 Set 按钮，进入 Velocity Inlet(速度边界设置)对话框：在 Turbulence Specification Method 下拉列表框中选择 Intensity and Hydraulic Diameter，在 Turbulence Intensity 文本框中输入 5，在 Hydraulic Diameter 文本框中输入 5.6，如图 6-8(a)所示。在 Boundary Conditions 对话框的 Phase 下拉列表框中选择 sewage，单击 Set 按钮，在 Velocity Inlet 对话框的 Velocity Magnitude 文本框中输入 0.019，如图 6-8(b)所示。在 Boundary Conditions 对话框的 Phase 下拉列表框中选择 sediments，单击 Set 按钮，在 Velocity Inlet 对话框的 Velocity Magnitude 文本框中输入 0.019，在 Volume Fraction 文本框中输入 0.034，如图 6-8(c)所示。

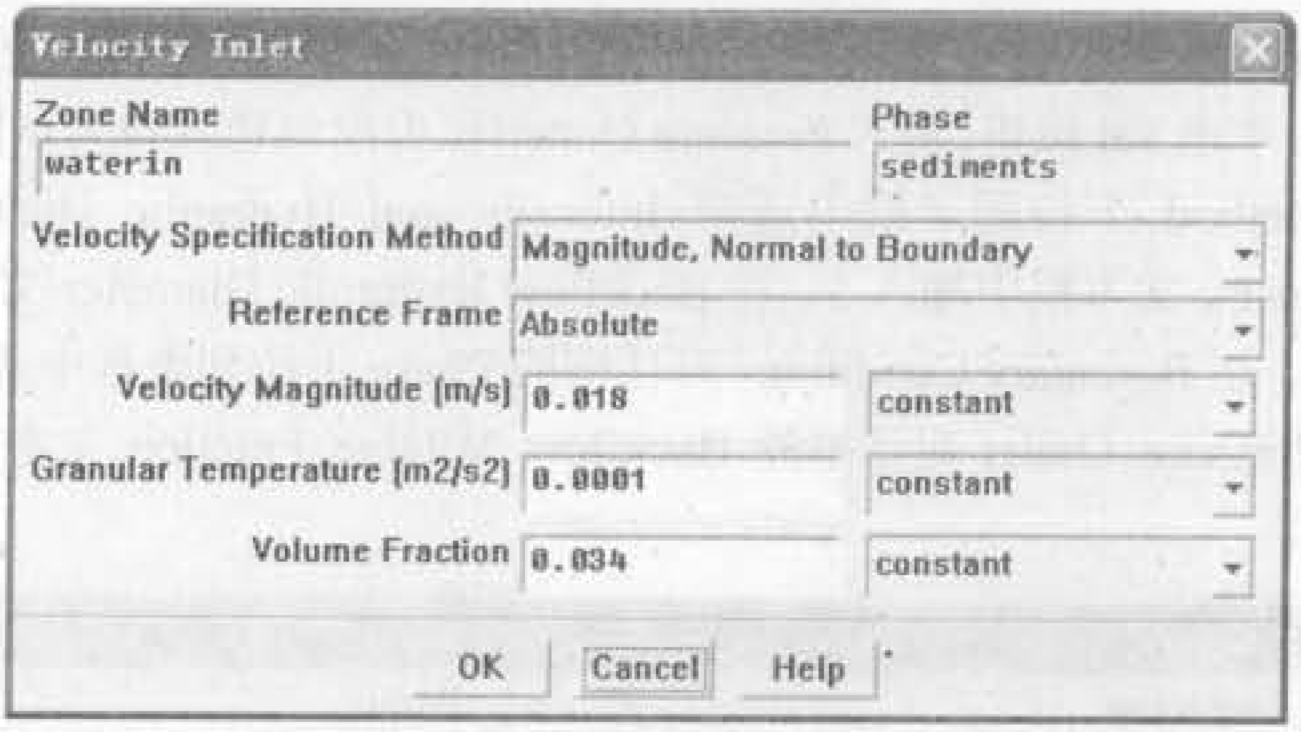


(a)



(b)

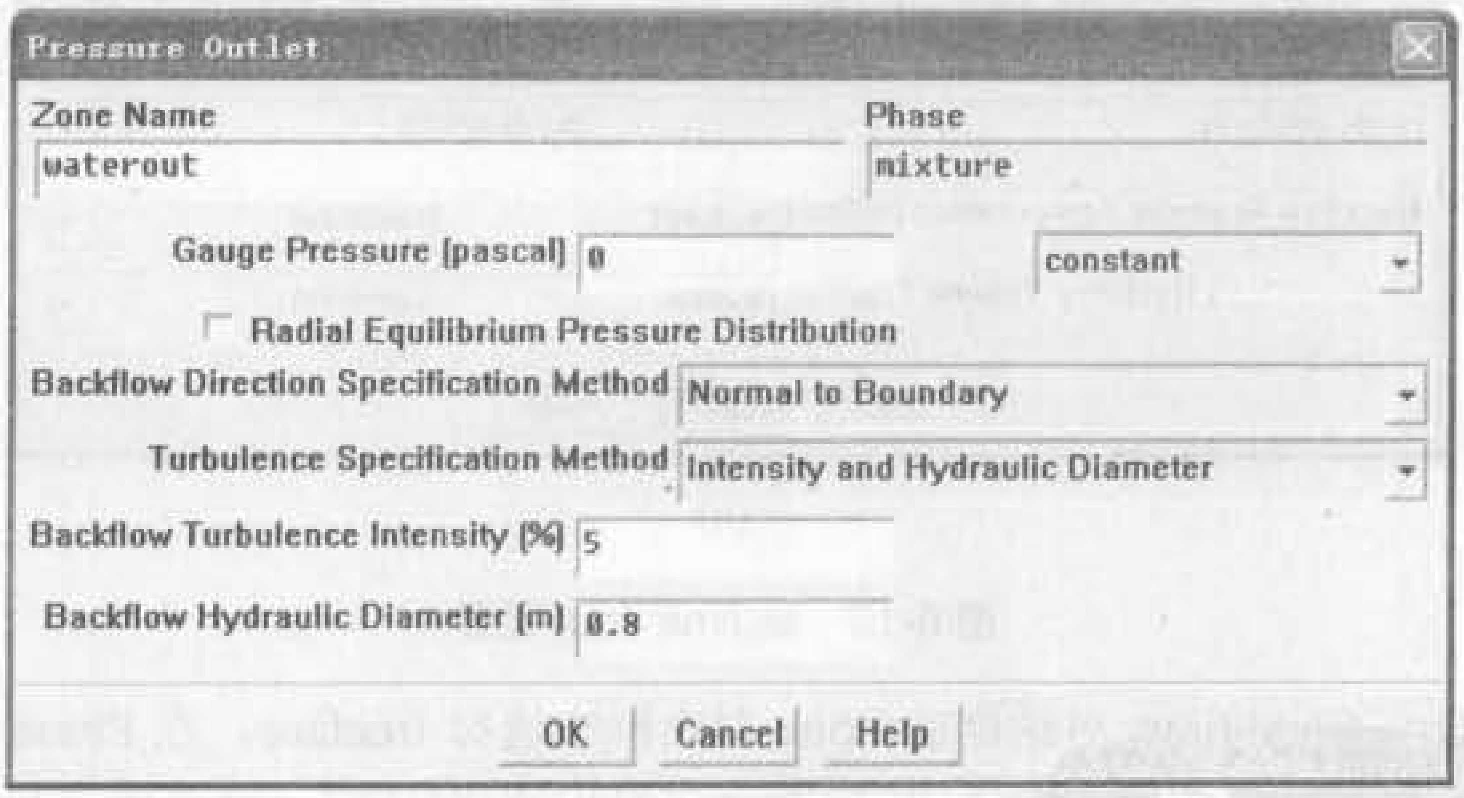
图 6-8 waterin 设置



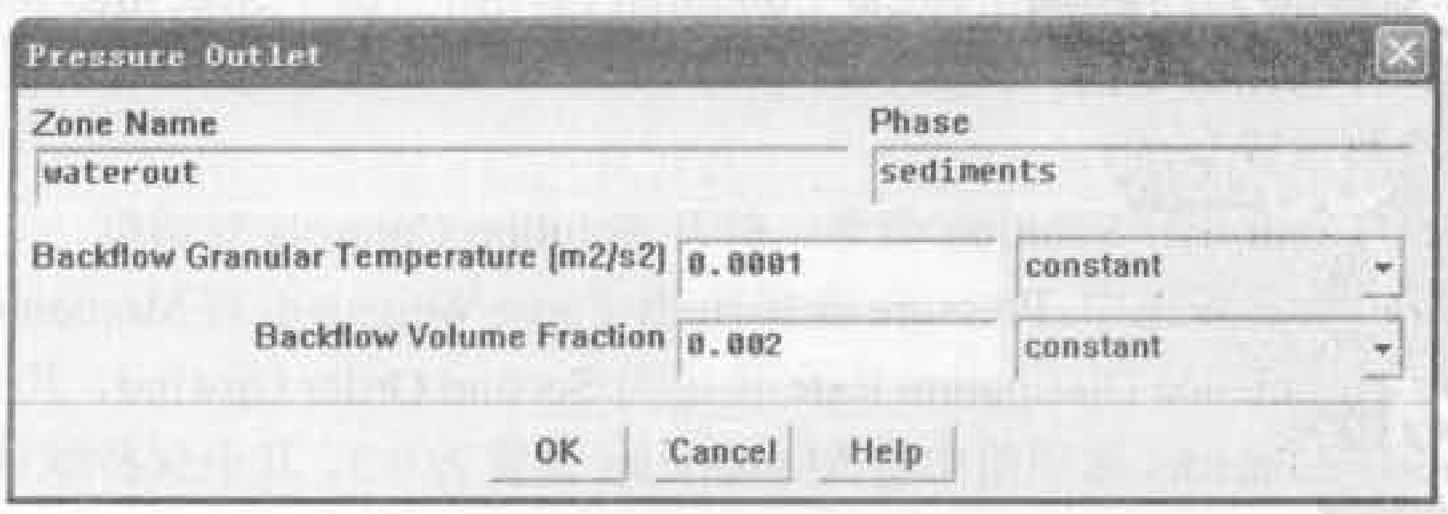
(c)

图 6-8　(续)

在 Boundary Conditions 对话框的 Zone 列表框中选择 waterout，在 Phase 下拉列表框中选择 mixture，单击 Set 按钮，进入 Pressure Outlet(压力出口边界设置)对话框；在 Turbulence Specification Method 下拉列表框中选择 Intensity and Hydraulic Diameter，在 Backflow Turbulence Intensity 文本框中输入 5，在 Backflow Hydraulic Diameter 文本框中输入 0.8(对于任意截面形状管道的当量直径，都可按截面积的四倍和截面周长之比计算)，如图 6-9(a)所示。在 Boundary Conditions 对话框的 Phase 下拉列表框中选择 sediments，单击 Set 按钮，在 Pressure Outlet 对话框的 Backflow Volume Fraction 文本框中输入 0.002，如图 6-9(b)所示。



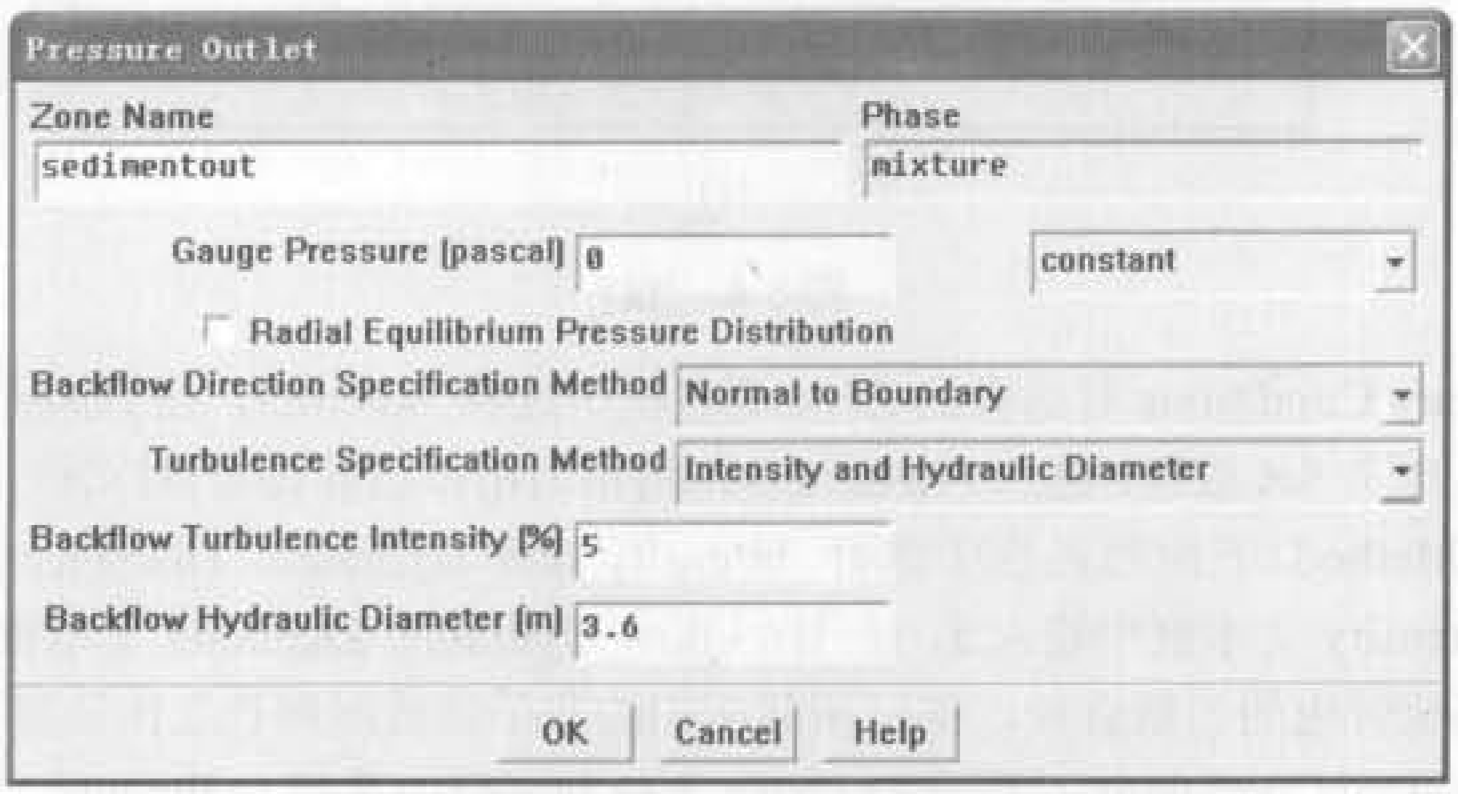
(a)



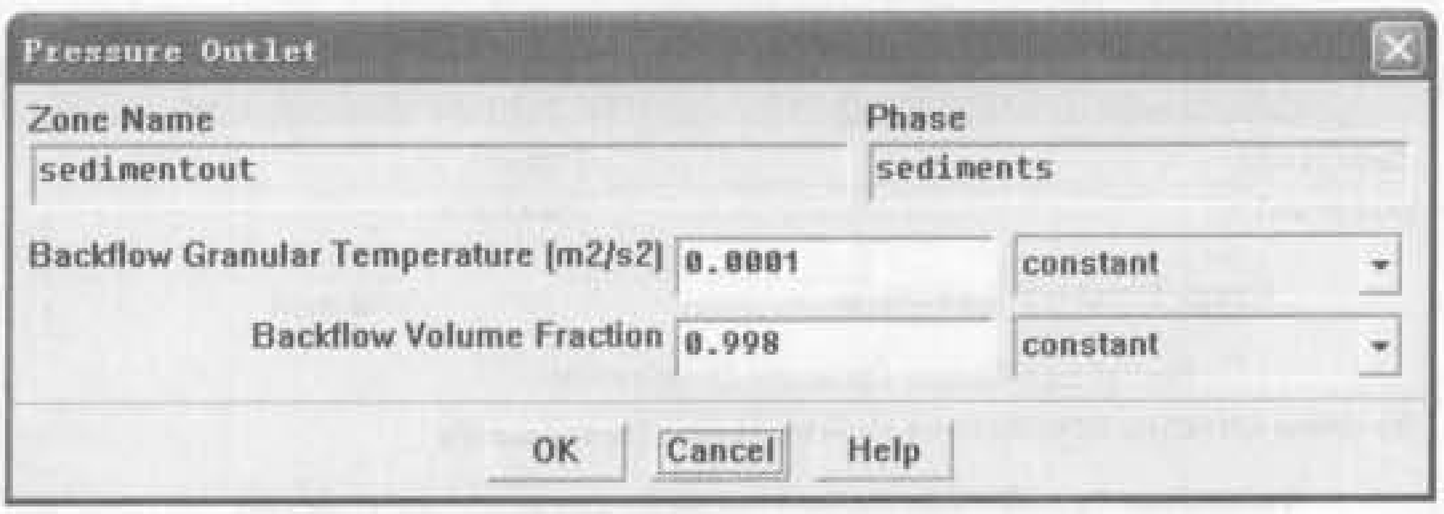
(b)

图 6-9　waterout 设置

在 Boundary Conditions 对话框的 Zone 列表框中选择 sedimentout，在 Phase 下拉列表框中选择 mixture，单击 Set 按钮，进入 Pressure Outlet(压力出口边界设置)对话框，在 Turbulence Specification Method 下拉列表框中选择 Intensity and Hydraulic Diameter，在 Backflow Turbulence Intensity 文本框中输入 5，在 Backflow Hydraulic Diameter 文本框中输入 3.6，如图 6-10(a)所示。在 Boundary Conditions 对话框的 Phase 下拉列表框中选择 sediments，单击 Set 按钮，在 Pressure Outlet 对话框的 Backflow Volume Fraction 文本框中输入 0.998，如图 6-10(b)所示。



(a)



(b)

图 6-10 sedimentout 设置

在 Boundary Conditions 对话框的 Zone 列表框中选择 freeface，在 Phase 下拉列表框中选择 mixture，单击 Set 按钮，进入 Wall(壁面边界设置)对话框。这个面是自由表面，这里以表面没有剪切力为自由表面，在 Shear Condition 选项组中选中 Specified Shear 单选按钮，其三个分量都保持默认的 0 值，如图 6-11 所示。

7) 求解参数设置

选择 Solve→Controls→Solution 命令，打开 Solution Controls 对话框。

在 Discretization 选项组中，Pressure 选择 Body Force Weighted，将 Momentum、Turbulence Kinetic Energy、Turbulence Dissipation Rate 设定为 Second Order Upwind，其余保持默认值。在 Under-Relaxation Factors 选项组中将 Momentum 设置为 0.3，其余保持默认值，如图 6-12 所示。

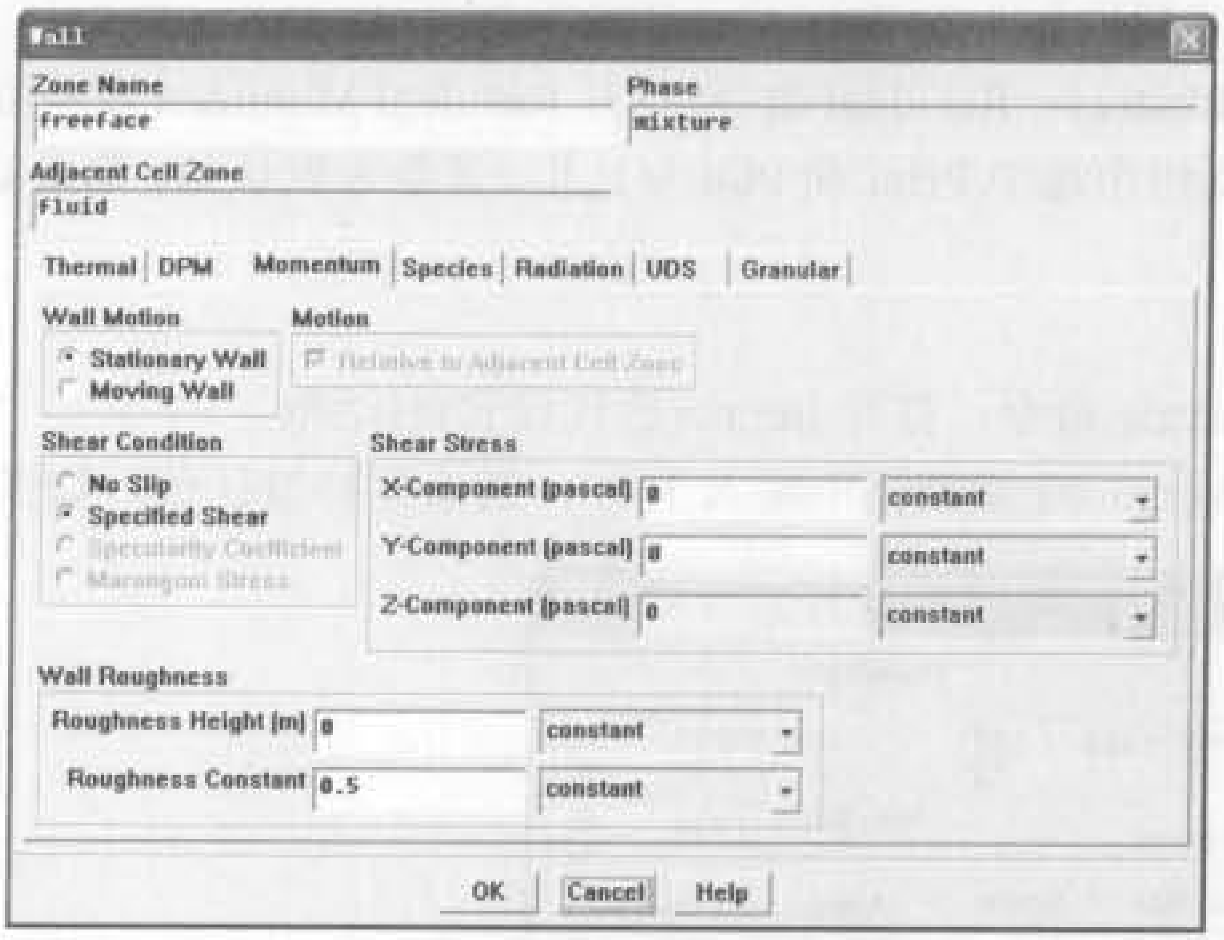


图 6-11　freeface 设置

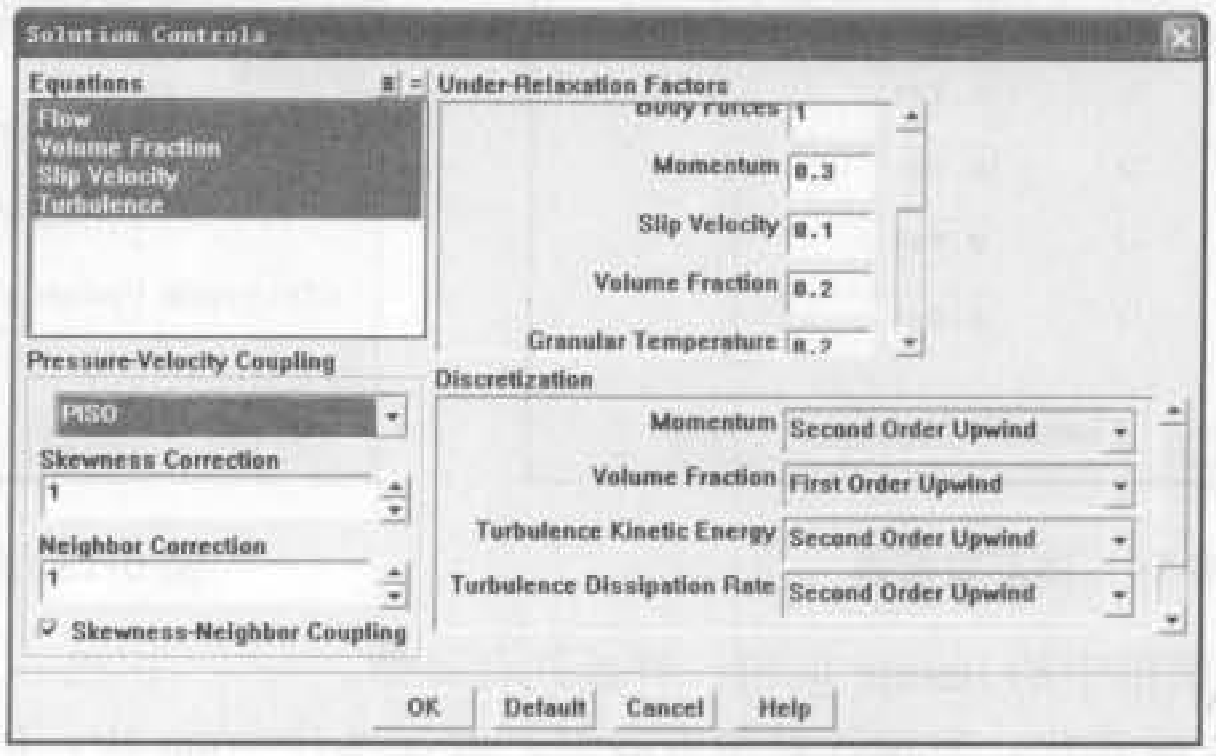


图 6-12　求解控制设置

8)　流场初始化

选择 Solve→Initializate→Initializate 命令，打开 Solution Initialization 对话框。

在 Compute From 下拉列表框中选择 all-zones，将 Initial Values 选项组中的 sediments Volume Fraction 设为 0，其余保持默认值，如图 6-13 所示。

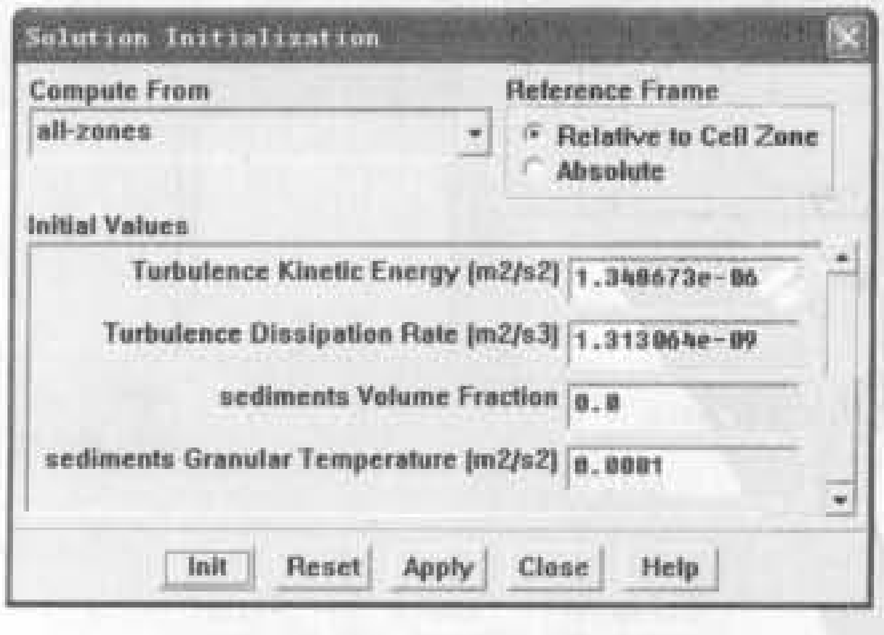


图 6-13　初始化

9) 设置残差监视器

选择 Solve→Monitors→Residual 命令，打开 Residual Monitors(残差监视器设置)对话框。

在 Options 选项组中选中 Print 和 Plot 复选框，其余为默认值，单击 OK 结束，如图 6-14 所示。

10) 迭代计算

选择 Solve→Iterate 命令，打开 Iterate(迭代设置)对话框。

在 Number of Iterations 微调框中输入 1000，其余保持默认值，如图 6-15 所示。

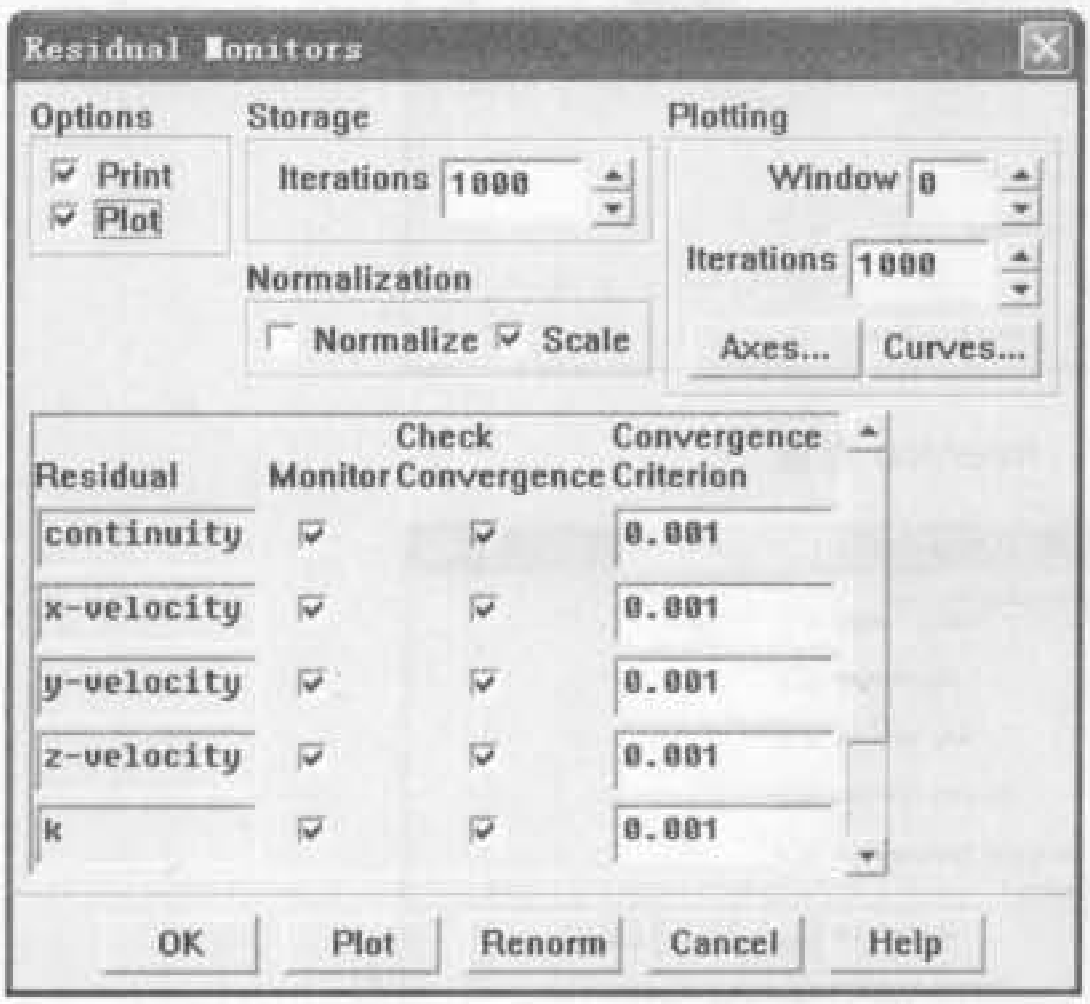


图 6-14 残差监视器

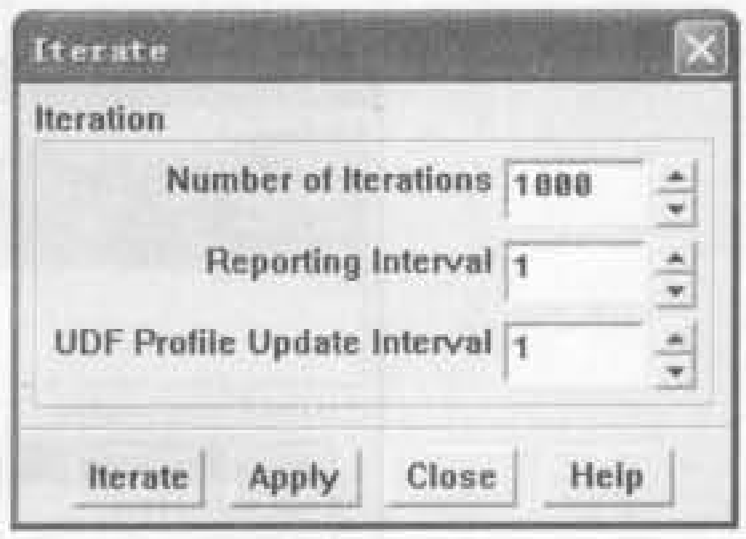


图 6-15 迭代设置

单击 Iterate 对话框中的 Iterate 按钮，开始迭代计算。

当计算完 1000 次迭代，发现计算没有收敛，继续迭代 2000 次，仍然没有收敛，但各残差曲线基本上在一直线附近振荡。选择 Report→Fluxes 命令打开 Flux Reports 对话框，如图 6-16 所示，查看进出口流量基本守恒了，近似认为计算达到要求。

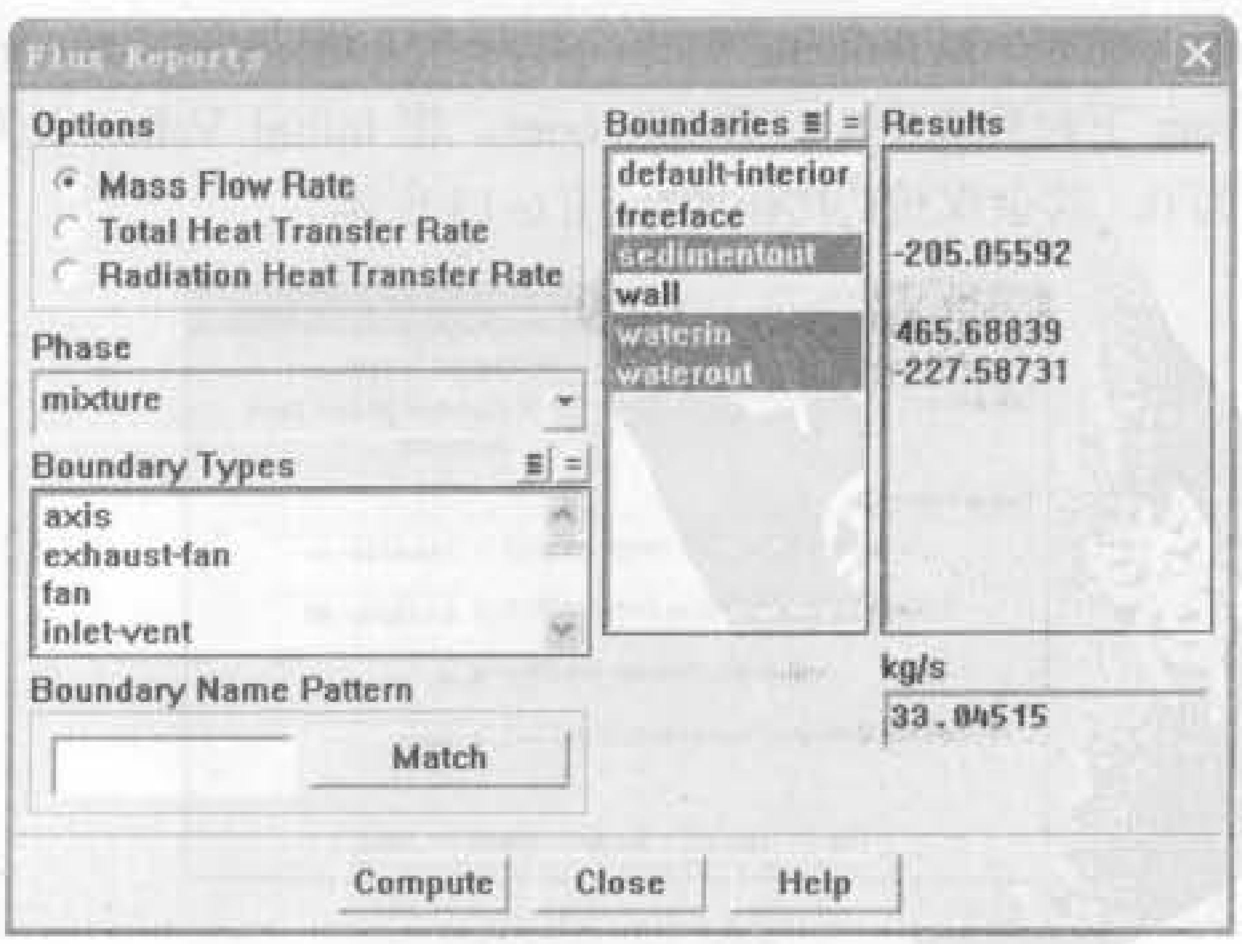


图 6-16 流量检验

4. 结果后处理

1)　云图显示

选择 Display→Contours 命令，打开 Contours 对话框(可参阅图 2-32)，选中 Filled。

在 Contours of 下拉列表框中选择 Pressure，在 Phase 下拉列表框中选择 mixture，单击 Display 按钮，显示如图 6-17 所示的压力云图。

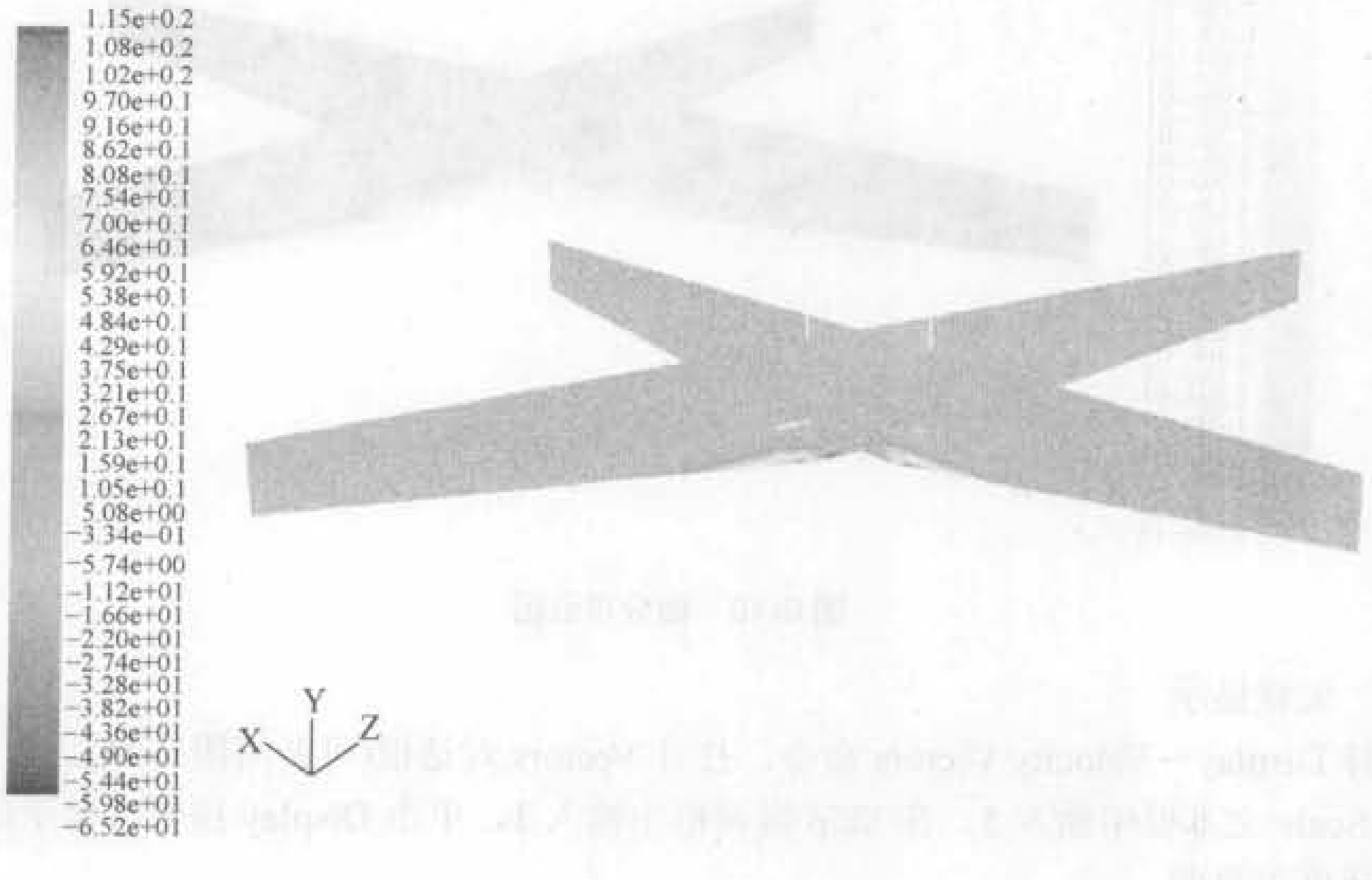


图 6-17　压力云图

在 Contours of 下拉列表框中选择 Velocity，在 Phase 下拉列表框中选择 mixture，单击 Display 按钮，显示如图 6-18 所示的速度云图。

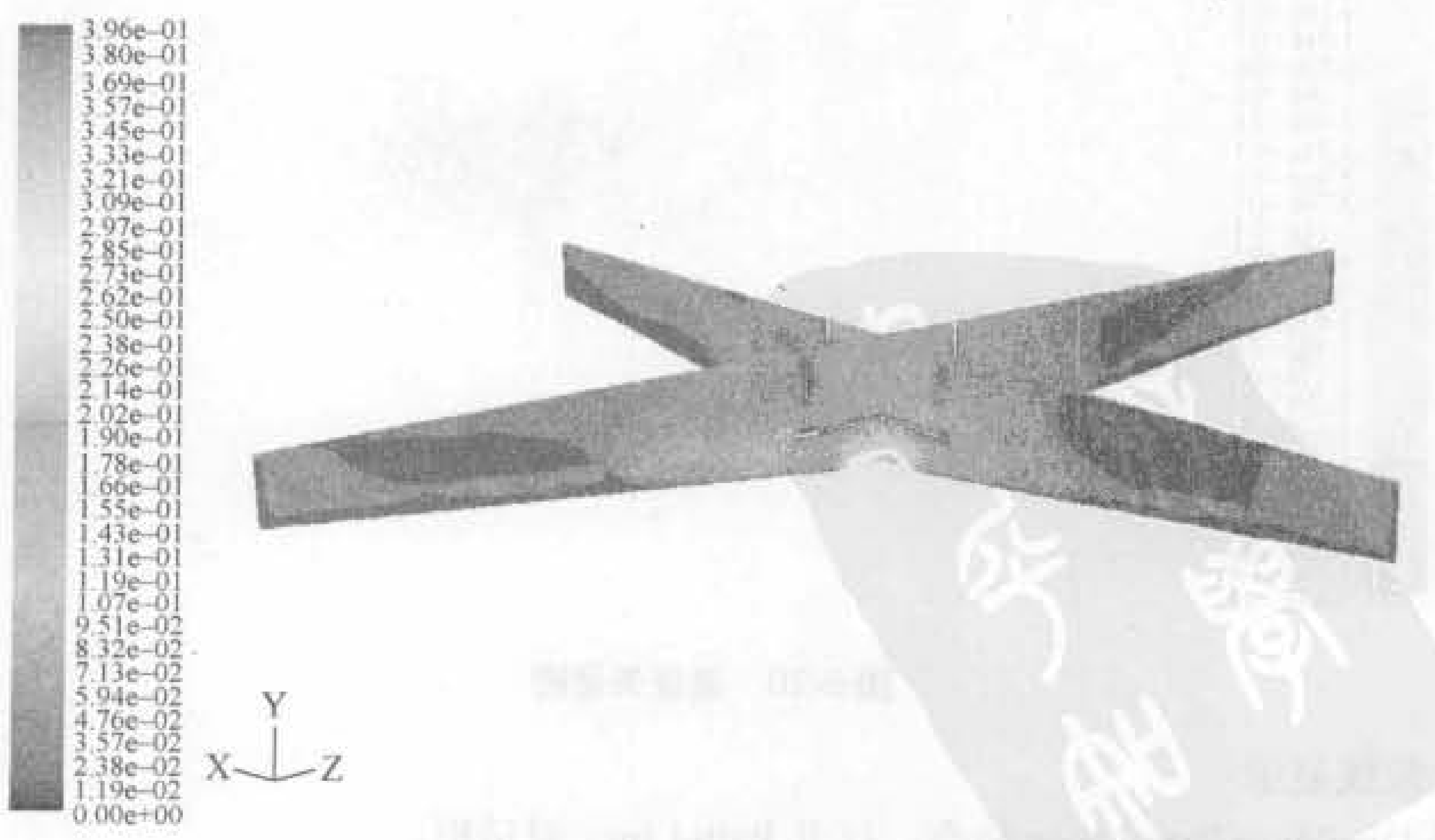


图 6-18　速度云图

在 Contours of 下拉列表框中选择 Phases，在 Phase 下拉列表框中选择 sediments，单击 Display 按钮，显示如图 6-19 所示的相分布云图。

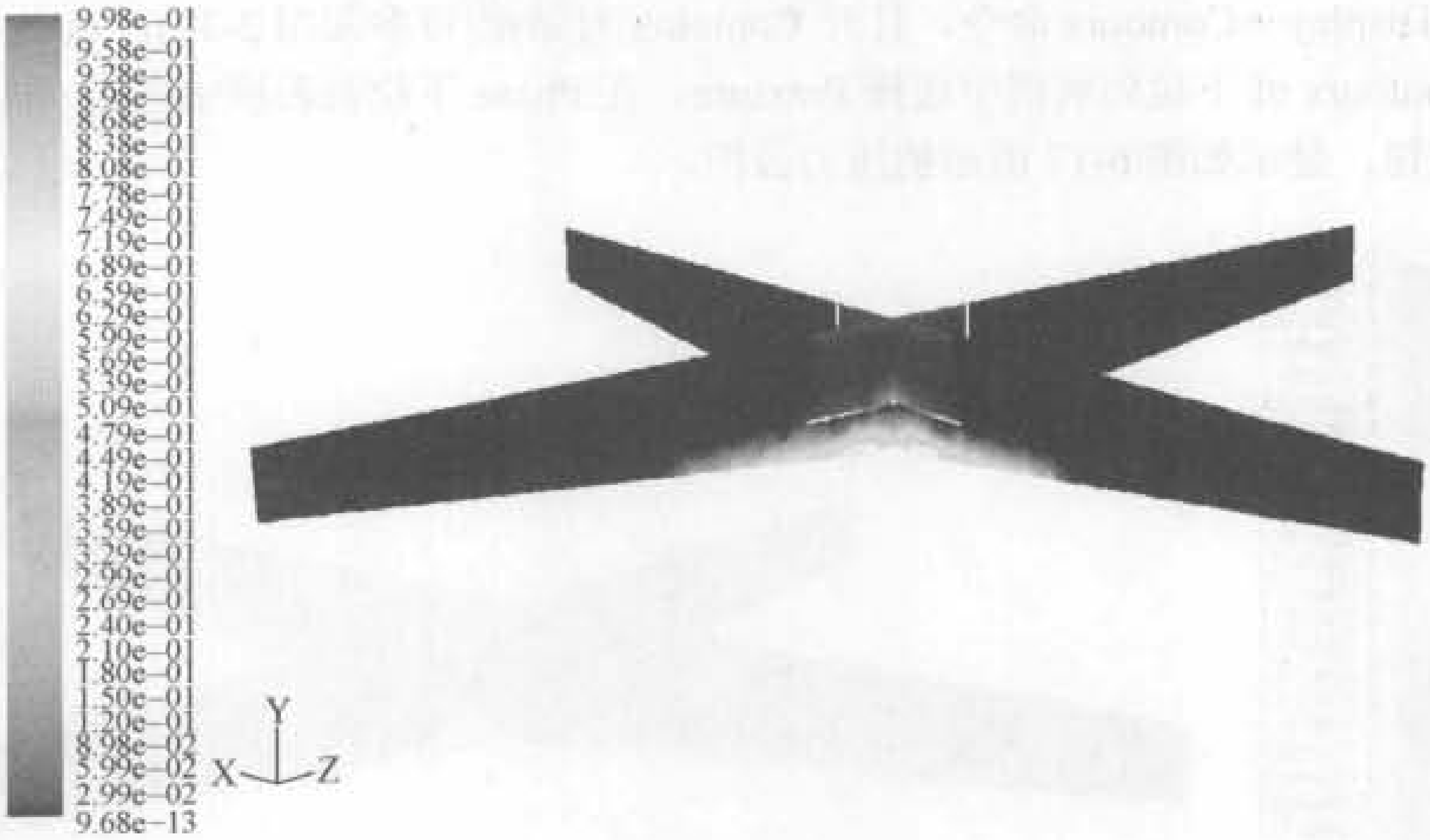


图 6-19 相分布云图

2) 矢量显示

选择 Display→Velocity Vectors 命令，打开 Vectors 对话框(可参阅图 2-36)。

在 Scale 文本框中输入 5，在 Skip 微调框中输入 1，单击 Display 按钮，显示如图 6-20 所示的速度矢量图。



图 6-20 速度矢量图

3) 流线显示

选择 Display→Path Lines 命令，打开 Path Lines 对话框。

在 Step 栏中输入 5000，在 Skip 栏中输入 1，在 Release from Surface 栏中选择 waterin，单击 Display 按钮，显示如图 6-21 所示的流线图。

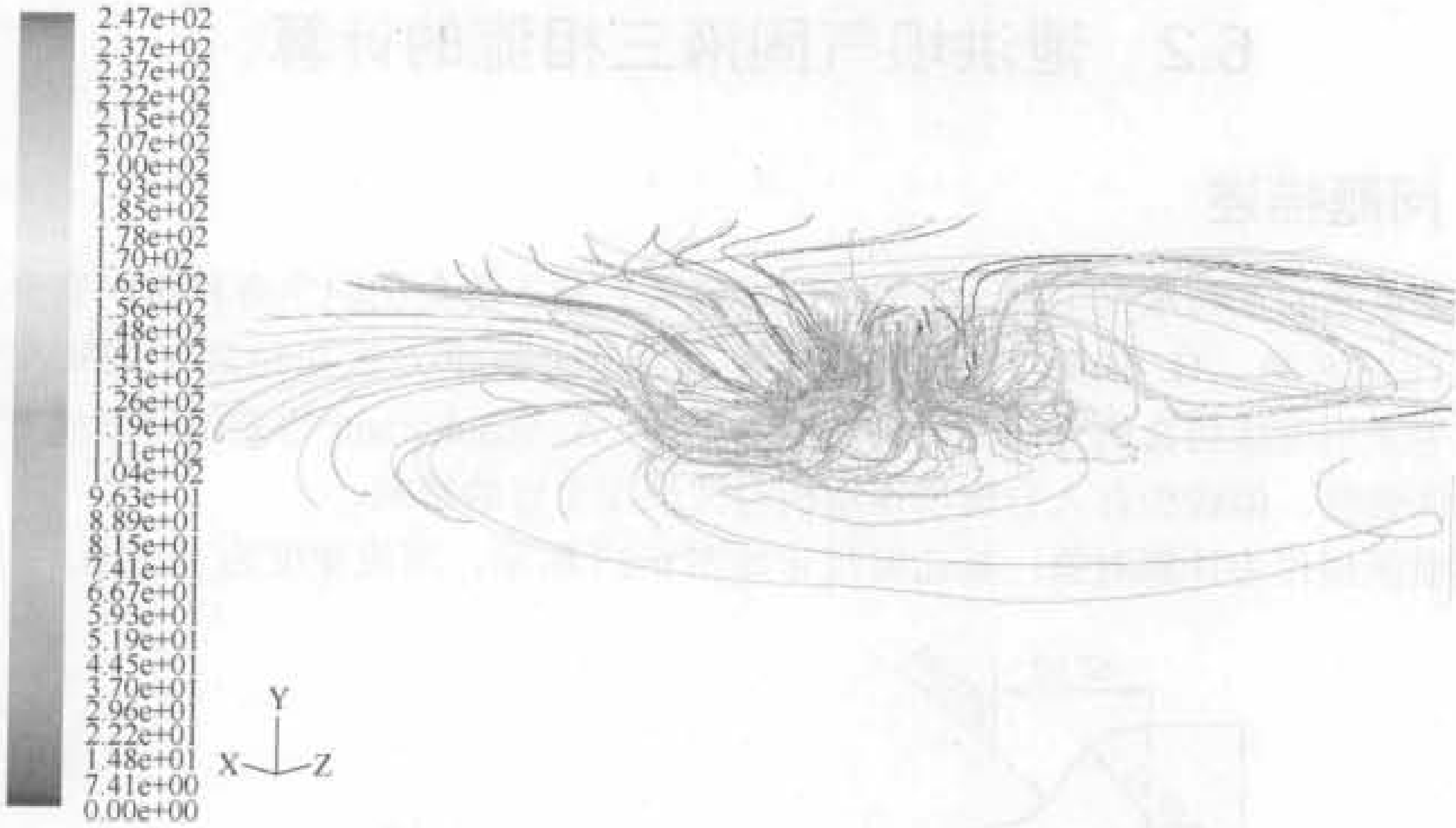


图 6-21　流线图

4)　XY 曲线显示

显示计算区域任意线段或任意面上的物理量的变化情况。这里在 sediments 面上创建一根线段。

选择 Surface→Line/Rake 命令，打开 Line/Rake Surface 对话框。输入 z0=-20，z1=20，其余坐标值输入 0，单击 Create 按钮，创建名为 line-7(该名称根据自己的情况命名，这里是默认名称)的线段。

选择 Plot→XY Plot 命令，打开 Solution XY Plot 对话框(可参阅图 2-38)。在 Y Axis Function 下拉列表框中选择 Phases，在 Phase 下拉列表框中选择 sewage，在 Surfaces 列表框中选择 line-7，单击 Plot 按钮，显示如图 6-22 所示的流线图。

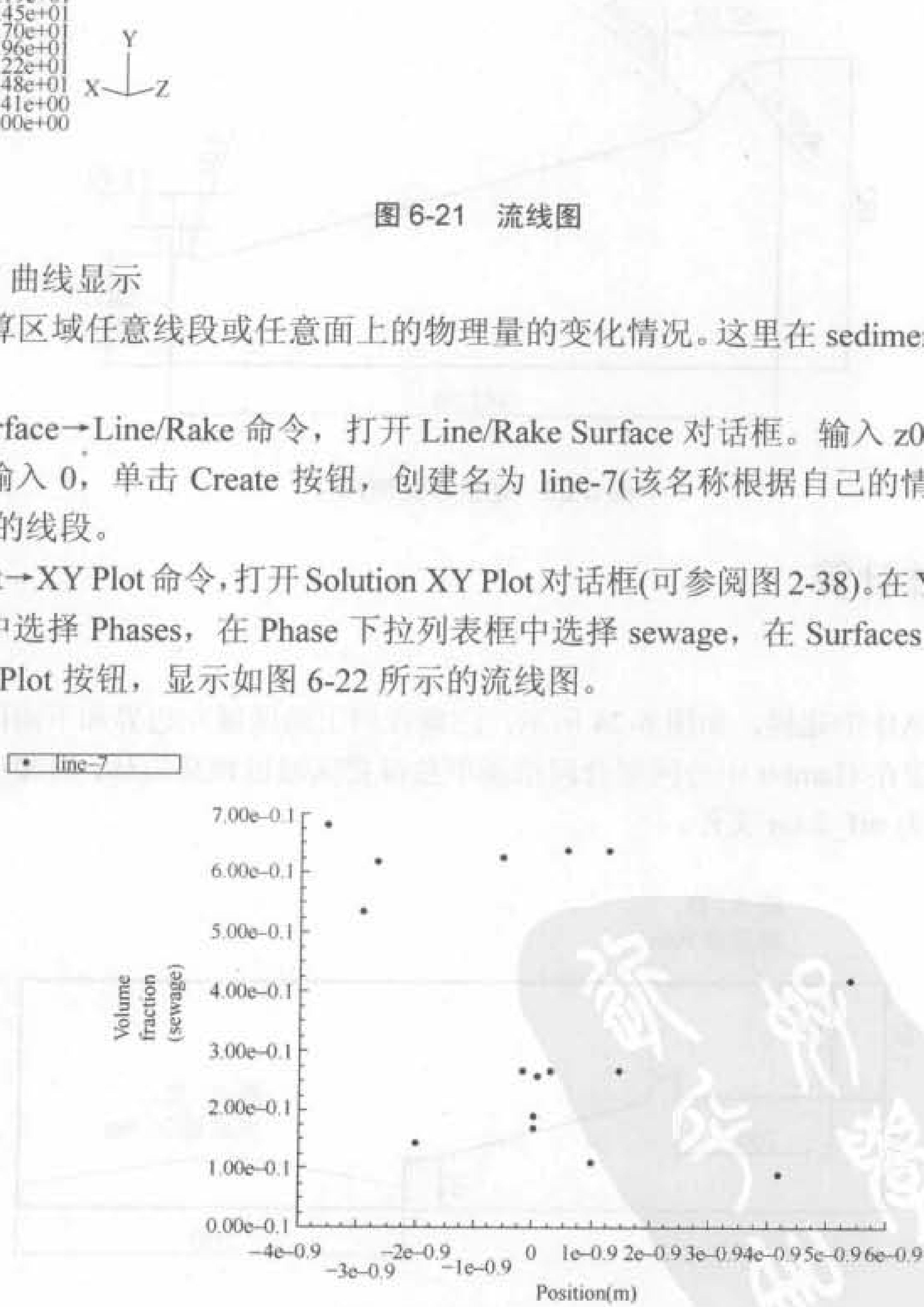


图 6-22　sediments 出口一线段上相分布曲线

6.2 泄洪坝气固液三相流的计算

6.2.1 问题描述

大坝是重要的水利水电工程，其安全性直接关系到下游人民的生命和财产安全，一直是人们关注的对象。H. Horii 等采用裂纹植入原理分析重力坝在地震中的安全性，S. Blazkova 等采用连续仿真的方法评价大坝的安全性，A. Shahkarami 等采用有限元方法分析了大坝坝体强度。但较少有人分析坝体结构对大坝安全性的影响。

以某泄洪坝作为计算对象，其几何尺寸如图 6-23 所示，河流宽度为 1150m。

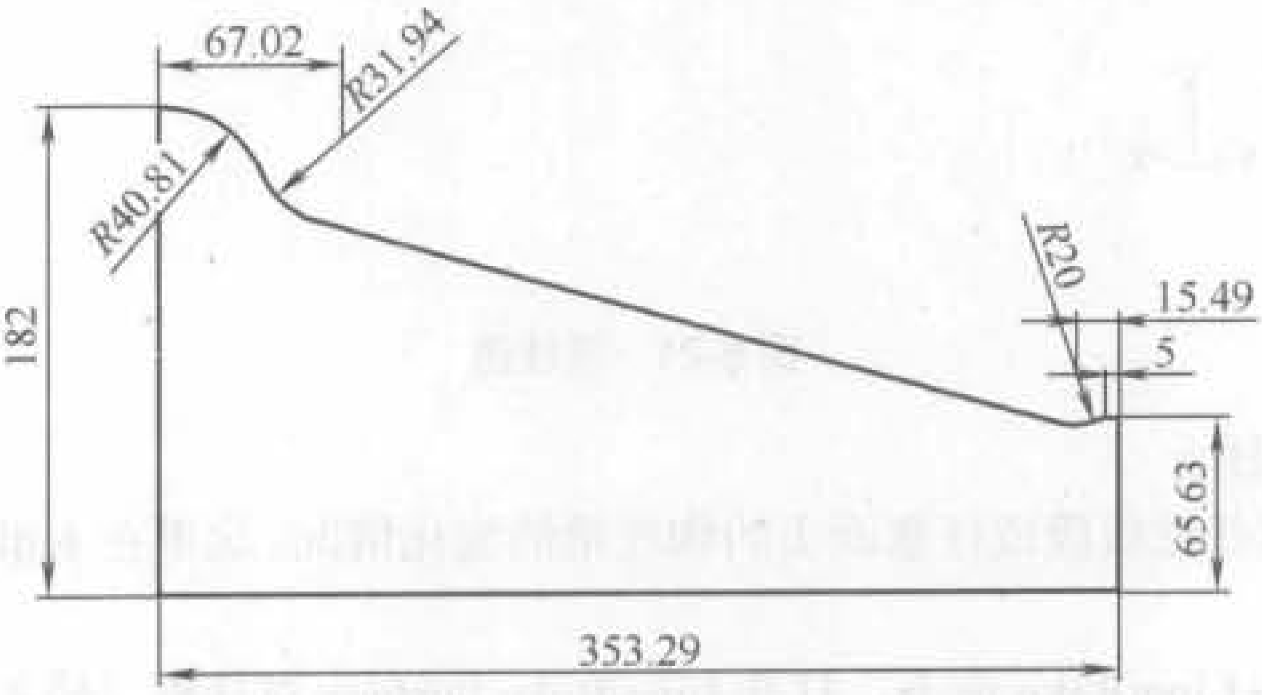


图 6-23 泄洪坝几何尺寸

6.2.2 具体计算

1. 建模

在 Auto CAD 中建模，如图 6-24 所示，注意在坝上游区域左边界和下游的左右边设置断点，为了方便在 Gambit 中分区划分网格及单独设置区域以预装流体，将线框 region 成区域，最后输出为 mf_2.sat 文件。

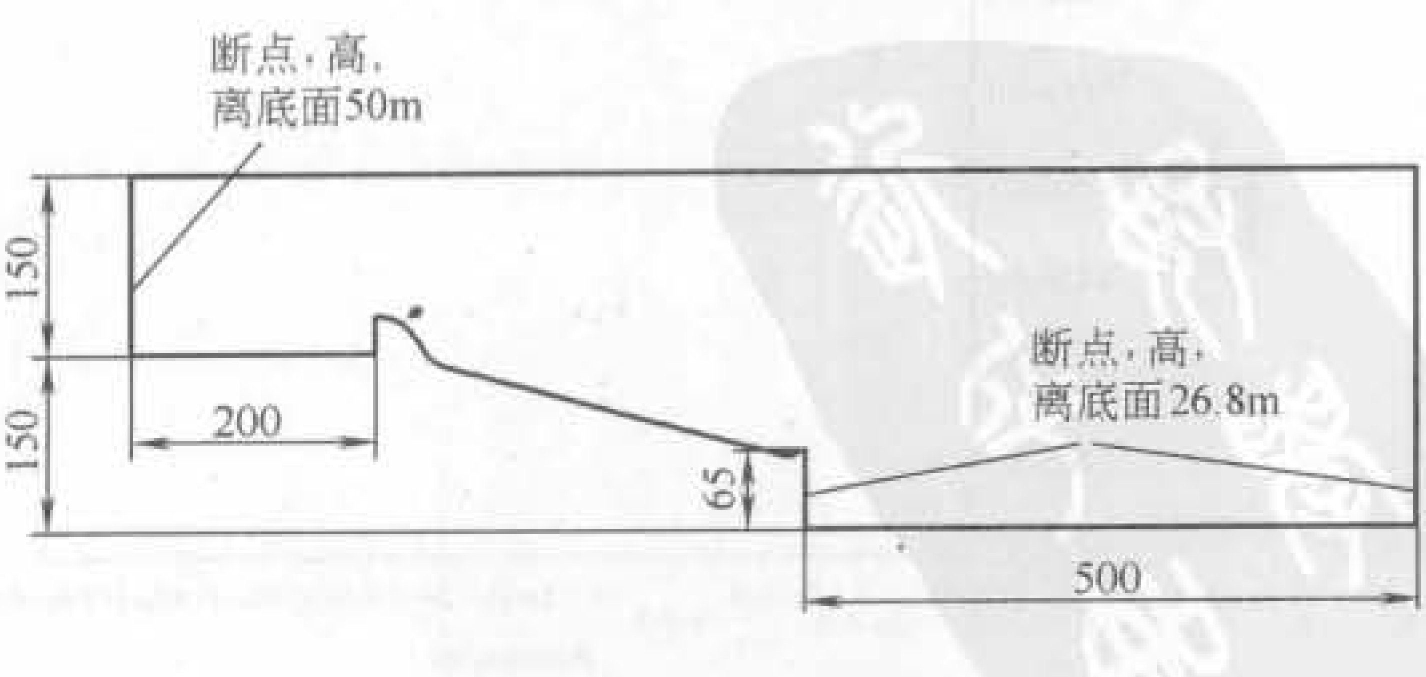


图 6-24 计算实体模型

2. 网格划分

将上述 mf_2.sat 文件导入到 Gambit 中，先删除面，创建一点(200,50)，分别连接线，而后构成三个面，如图 6-25 所示。

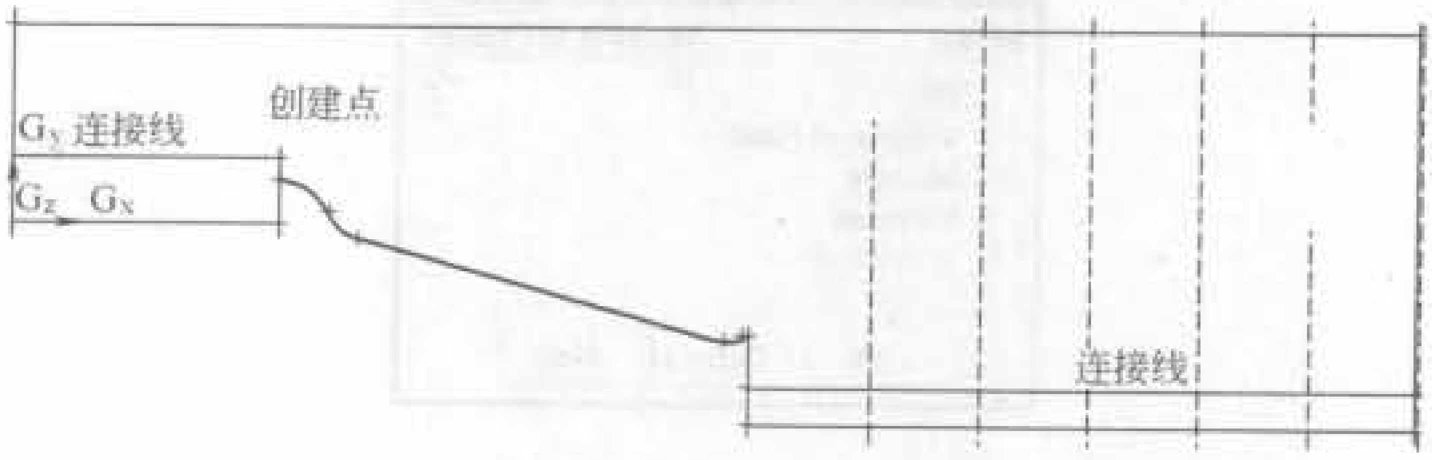


图 6-25　几何重新处理

因要观察流体在坝面的瞬态流动，坝面的网格要密些，将坝面的线先划分，interval size 设为 2，面积最大的面采用三角形网格划分，在 Mesh Volumes 对话框中，Elements 选择 Tri，Type 选择 Pave，interval size 设置为 10，共划分出 9389 个节点，18366 个单元；其余的两个面采用四边形网格划分，在 Mesh Volumes 对话框中，Elements 选择 Quad，Type 选择 Map，interval size 设置为 10，左下边的面共划分出 126 个节点，100 个单元，右下边的面共划分出 204 个节点，150 个单元。网格模型如图 6-26 所示。

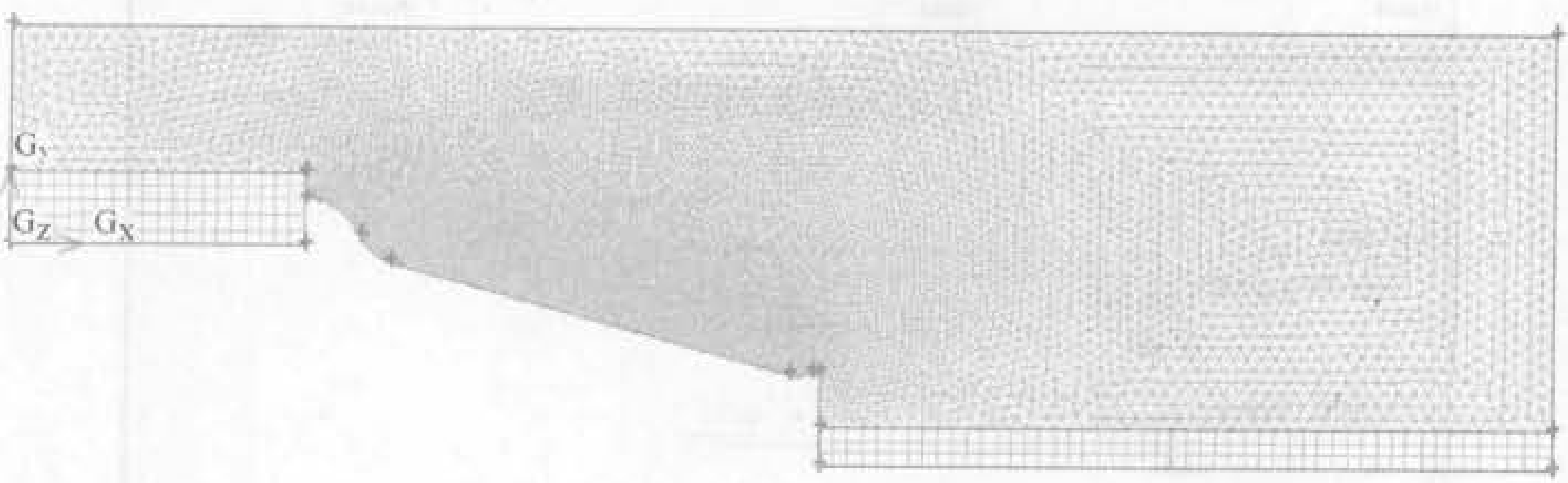


图 6-26　网格模型

在 Specify Boundary Types 对话框中，左边线段的下段设置为 VELOCITY_INLET 边界，命名为 waterin；右边线段的下段设置为 VELOCITY_INLET 边界，命名为 waterout；上边的线段和左右线段的上段设置为 PRESSURE_INLET 边界，命名为 airin。在 Specify Continuum Type 对话框中，左下边的小面设置为 FLUID，命名为 water1；右下边的小面设置为 FLUID，命名为 water2。保存后输出为 msh 文件，命名为 mf_2.msh。

3. Fluent 设置

1)　读入 msh 文件

选择 File→Read→Case 命令，读入 Gambit 输出的 mf_2.msh 文件，并检查网格(选择 Grid→Check 命令)，注意最小体积要大于零。应用 Grid→Smooth/Swap 命令消除网格的扭曲。

2)　模型设置

选择 Define→Models→Solver 命令，打开 Solver 对话框，在 Time 栏中选中 Unsteady，

保持默认设置。

选择 Define→Models→Multiphase 命令，打开 Multiphase Model 对话框，选择 Eulerian，Number of Phases 设置为 3，如图 6-27 所示。

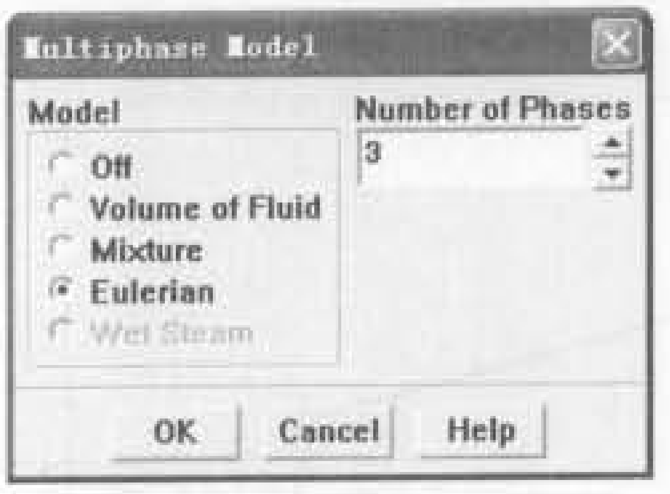


图 6-27　启动多相流模型

选择 Define→Models→Viscous 命令，打开 Viscous Model 对话框，选择 k-epsilon，其余保持默认设置。

3)　材料设置

选择 Define→Materials 命令，打开 Materials 对话框，如图 6-28 所示，分别从材料库中复制 air 和 water 材料，创建 sand 材料，密度为 2500kg/m^3，粘度为 10 kg/(m·s)。

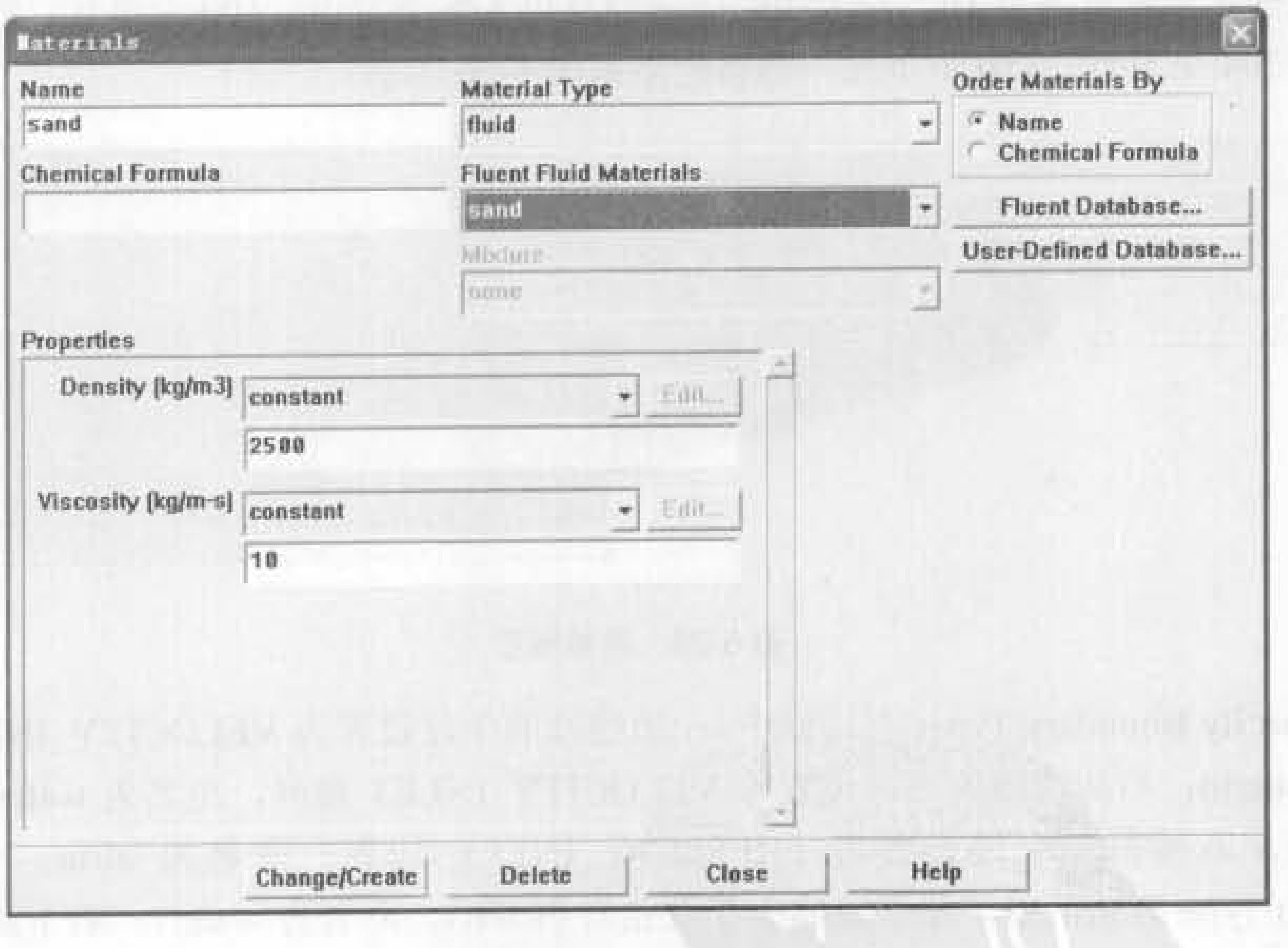


图 6-28　材料创建

4)　相设置

选择 Define→Phases 命令，打开 Phases 对话框，将 air 设置为主相，将 water 和 sand 设置为次相。在 sand 相时，选中 Granular，在 Diameter 栏的文本框中输入 0.0005，在 Granular Viscosity 栏的文本框中输入 10，在 Packing Limit 栏的文本框中输入 0.3，如图 6-29 所示。

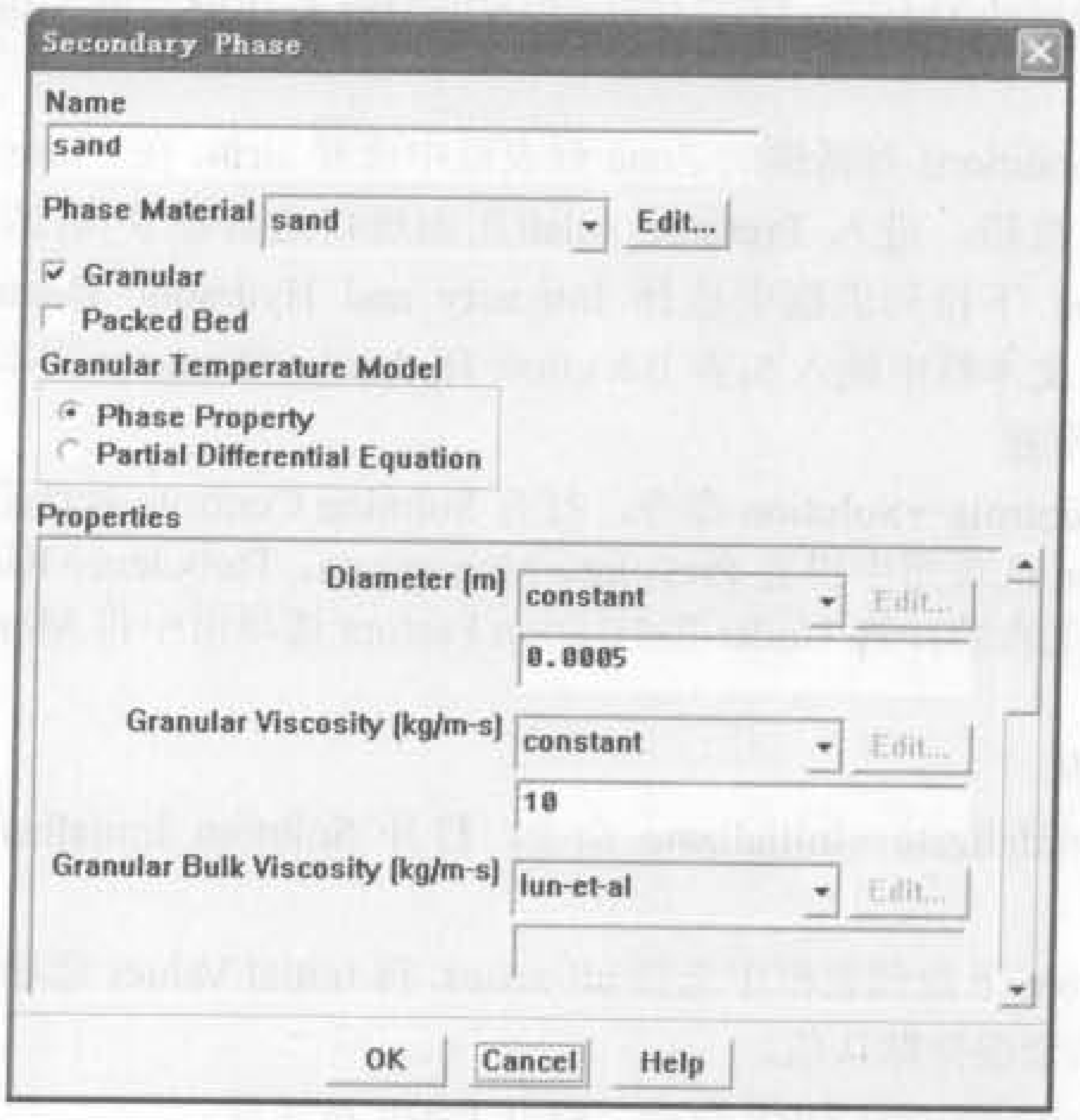


图 6-29　材料创建

5)　操作条件设置

选择 Define→Operating Conditions 命令，打开 Operating Conditions 对话框，设置参考压力的位置为 X=0，Y=80，选中 Gravity，设置 Y 方向的重力加速度为-9.81m/s^2。

6)　边界条件设置

选择 Define→Boundary Conditions 命令，打开 Boundary Conditions 对话框。

在 Zone 列表框中选择 waterin，在 Phase 下拉列表框中选择 mixture，单击 Set 按钮，进入 Velocity Inlet(速度边界设置)对话框；在 Turbulence Specification Method 下拉列表框中选择 Intensity and Hydraulic Diameter，在 Turbulence Intensity 文本框中输入 5，在 Hydraulic Diameter 文本框中输入 95.83(对于任意截面形状管道的当量直径，都可按截面积的四倍和截面周长之比计算)。在 Boundary Conditions 对话框的 Phase 下拉列表框中选择 water，单击 Set 按钮，在 Velocity Inlet 对话框的 Velocity Magnitude 文本框中输入 1，在 Volume Fraction 文本框中输入 0.9。在 Boundary Conditions 对话框的 Phase 下拉列表框中选择 sand，单击 Set 按钮，在 Velocity Inlet 对话框的 Velocity Magnitude 文本框中输入 0.99，在 Volume Fraction 文本框中输入 0.1。

在 Boundary Conditions 对话框的 Zone 列表框中选择 waterout，在 Phase 下拉列表框中选择 mixture，单击 Set 按钮，进入 Pressure Outlet(压力出口边界设置)对话框中；在 Turbulence Specification Method 下拉列表框中选择 Intensity and Hydraulic Diameter，在 Backflow Turbulence Intensity 文本框中输入 5，在 Backflow Hydraulic Diameter 文本框中输入 1039.98。在 Boundary Conditions 对话框的 Phase 下拉列表框中选择 water，单击 Set 按钮，在 Pressure Outlet 对话框的 Velocity Magnitude 文本框中输入-1.86，在 Volume Fration 文本框中输入 0.96。在 Boundary Conditions 对话框的 Phase 下拉列表框中选择 sand，单击 Set 按钮，在

Pressure Outlet 对话框的 Velocity Magnitude 文本框中输入-1.85，在 Volume Fration 文本中输入 0.04。

在 Boundary Conditions 对话框的 Zone 列表框中选择 airin，在 Phase 下拉列表框中选择 mixture，单击 Set 按钮，进入 Pressure Inlet(压力进口边界设置)对话框，在 Turbulence Specification Method 下拉列表框中选择 Intensity and Hydraulic Diameter，在 Backflow Turbulence Intensity 文本框中输入 5，在 Backflow Hydraulic Diameter 文本框中输入 1099.53。

7) 求解参数设置

选择 Solve→Controls→Solution 命令，打开 Solution Controls 对话框(可参阅图 6-12)。

在 Discretization 选项组中设置 Pressure、Momentum、Turbulence Kinetic Energy 等参数(参考 6.1 节中的相关内容)，在 Under-Relaxation Factors 选项组中将 Momentum 设置为 0.3，其余保持默认值。

8) 流场初始化

选择 Solve→Initializate→Initializate 命令，打开 Solution Initialization 对话框(可参阅图 6-13)。

在 Compute From 下拉列表框中选择 all-zones，将 Initial Values 选项组中的 sand Volume Fraction 设为 0，其余保持默认值。

选择 Solve→Initializate→Patch 命令，打开 Patch 对话框。

在 Zones to Patch 列表框中选择 water1 和 water2，在 Phase 下拉列表框中选择 water，在 Variable 中选择 Volume Fraction，在 Value 文本框中输入 1，单击 Patch 按钮，如图 6-30 所示。

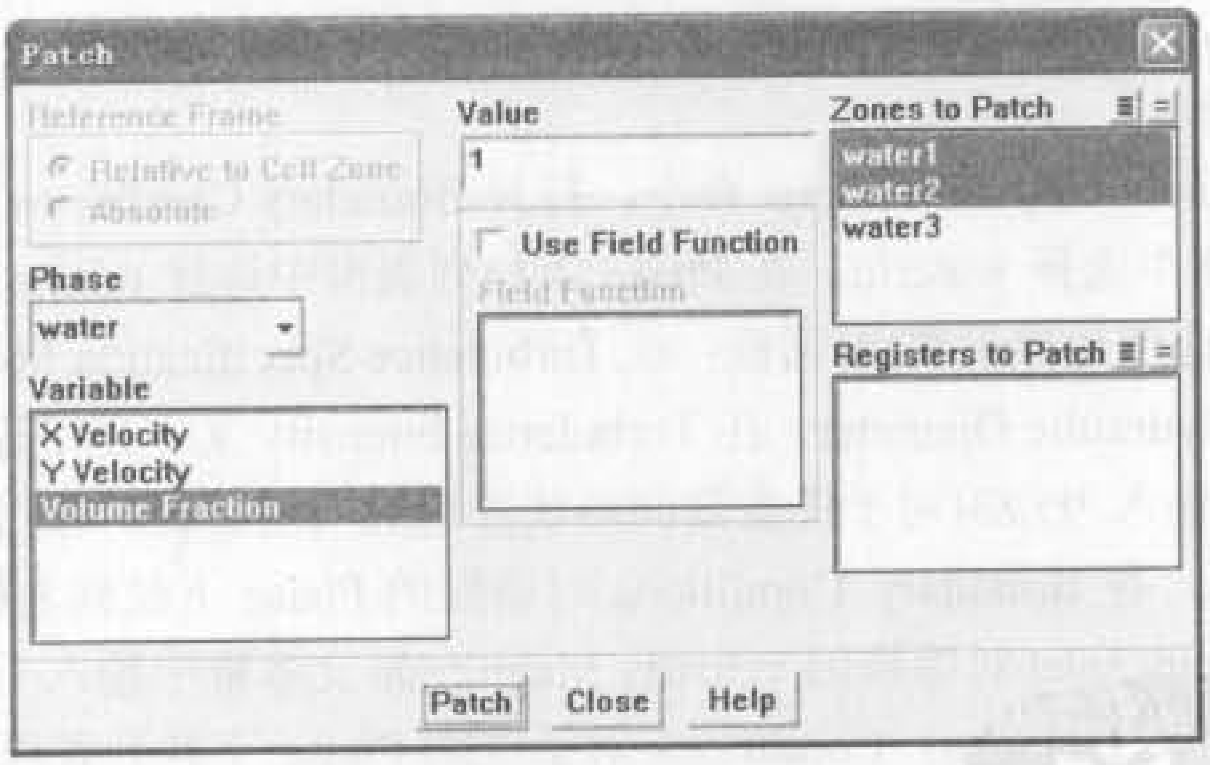


图 6-30 预装水设置

9) 设置残差监视器

选择 Solve→Monitors→Residual 命令，打开 Residual Monitors 对话框(可参阅图 6-14)。

在 Opions 选项组中选中 Plot 复选框，其余保持默认值，单击 OK 按钮结束。

10) 动画记录

选择 Solve→Animate→Define 命令，打开 Solution Animation 对话框。

单击 Animation Sequences 右边的上箭头，即增加 1 个动画记录，下面出现一亮显行，单击后面的 Define 按钮，出现 Animation Sequence 对话框；在 Name 文本框中输入 mf_2，在 Display Type 中选择 Contours，出现 Contours 对话框；在 Contours of 下拉列表框中选择

Phase，在 Phase 下拉列表框中选择 water，在 Options 选项组中选中 Filled 复选框，其余保持默认值，单击 Display 按钮，图形窗口出现计算区域初始的相分布，如图 6-31 所示。

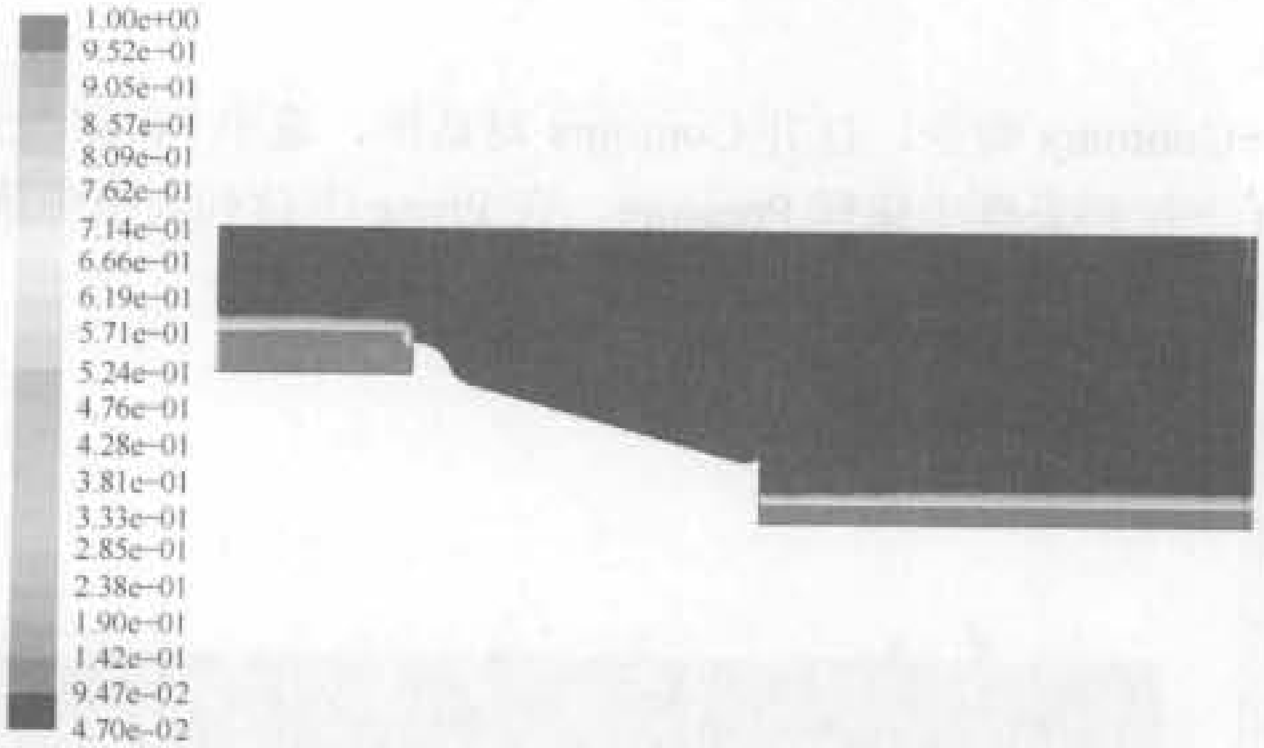


图 6-31　初始相分布

单击 Close 关闭 Contours 对话框，单击 OK 关闭 Animation Sequence 对话框，在 Solution Animation 对话框中的 Every 文本框中输入 10，在 When 栏中选中 Time Step，单击 OK 关闭 Solution Animation 对话框。

11) 迭代计算

选择 Solve→Iterate 命令，打开 Iterate 对话框。

在 Time Step Size 文本框中输入 0.01，在 Number of Time Steps 微调框中输入 50000，在 Max Iterations per Time Step 微调框中输入 40，其余保持默认值，如图 6-32 所示。

选择 File→Write→Autosave 命令，打开 Autosave Cas/Data 对话框；在 Autosave Case File Frequency 和 Autosave Case File Frequency 栏中均输入 500，表示 500 步后自动保存一次 case 和 dat 文件；单击 OK 按钮关闭该对话框。

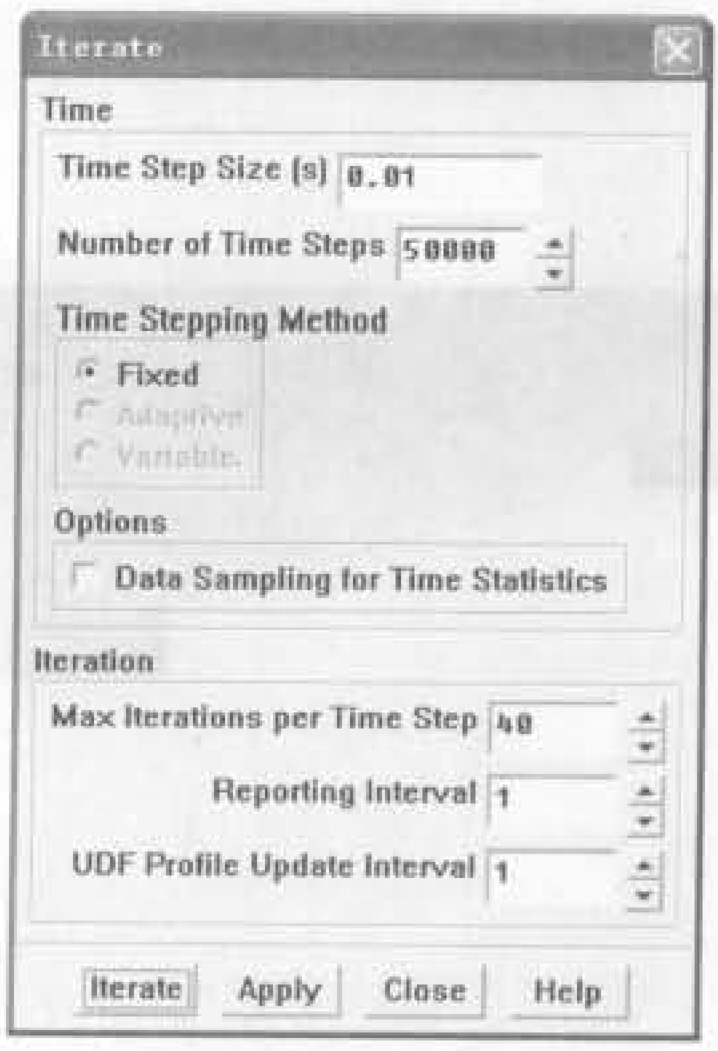


图 6-32　迭代设置

单击 Iterate 对话框中的 Iterate 按钮，开始迭代计算。

4. 结果后处理

1) 云图显示

选择 Display→Contours 命令，打开 Contours 对话框，选中 Filled 复选框。

在 Contours of 下拉列表框中选择 Pressure，在 Phase 下拉列表框中选择 mixture，单击 Display 按钮，显示如图 6-33 所示的压力云图。

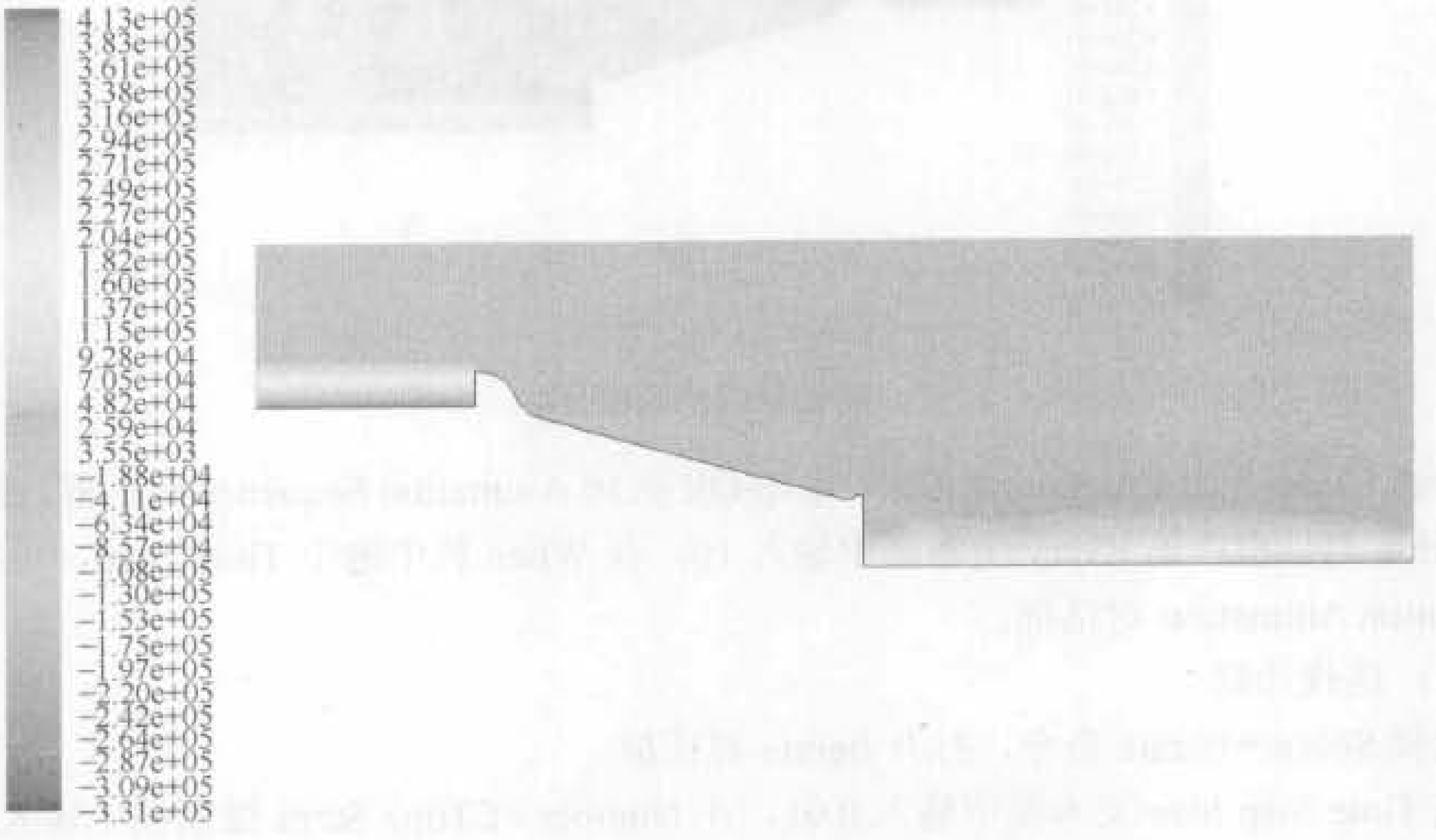


图 6-33 压力云图

在 Contours of 下拉列表框中选择 Velocity，在 Phase 下拉列表框中选择 water，单击 Display 按钮，显示如图 6-34 所示的速度云图。



图 6-34 速度云图

在 Contours of 下拉列表框中选择 Phases，在 Phase 下拉列表框中选择 sediments，单击 Display 按钮，显示如图 6-35 所示的相分布云图。

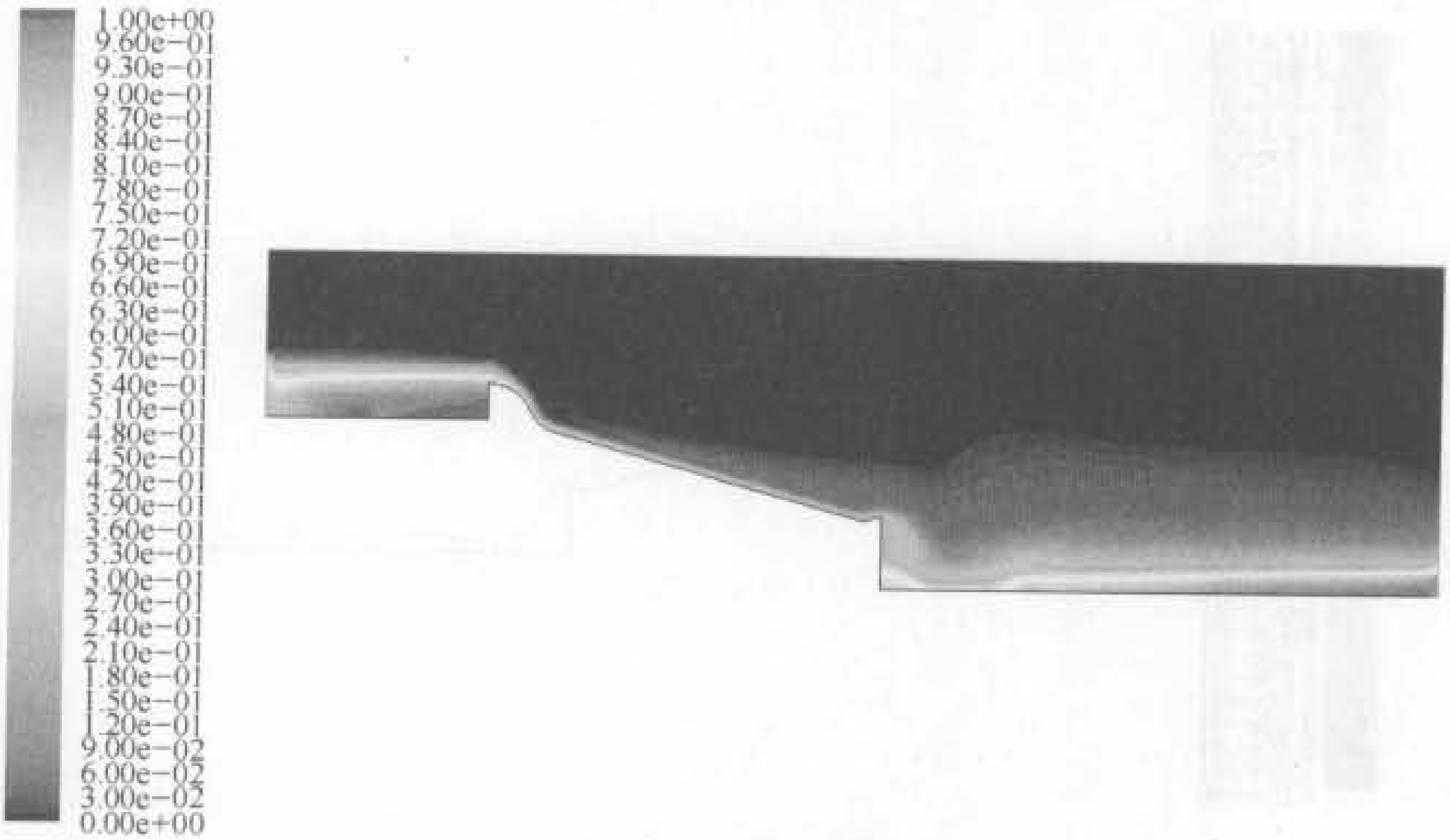


图 6-35　相分布云图

2)　矢量显示

选择 Display→Velocity Vectors 命令，打开 Vectors 对话框。

在 Vectors of 下拉列表框中选择 Velocity，在 Phase 下拉列表框中选择 water，在 Scale 文本框中输入 10，在 Skip 微调框中输入 1，单击 Display 按钮，显示如图 6-36 所示的速度矢量图。

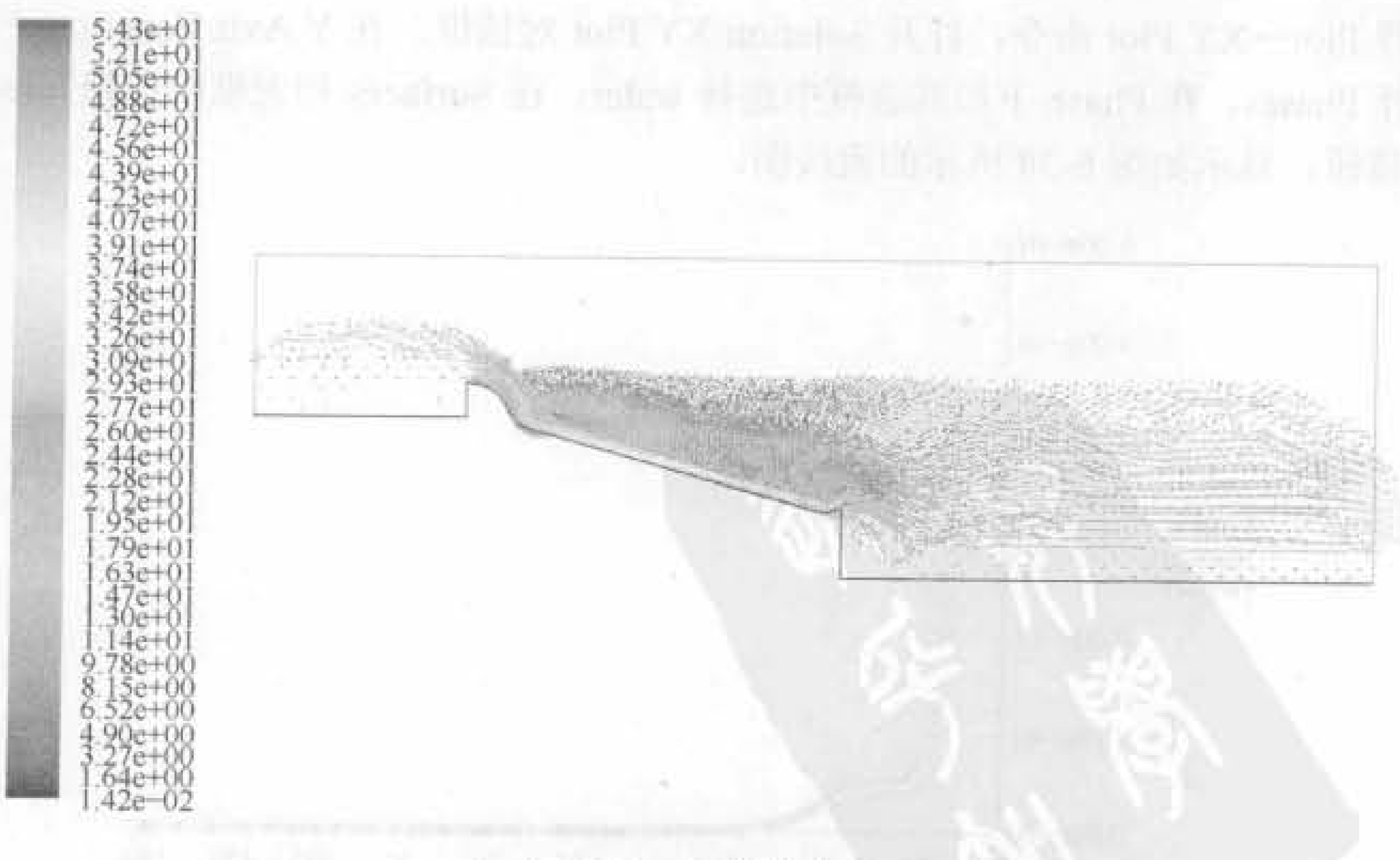


图 6-36　速度矢量图

3)　流线显示

选择 Display→Path Lines 命令，打开 Path Lines 对话框。

在 Step 栏中输入 120000，在 Skip 栏中输入 0，在 Release from Surface 栏中选择 waterin，单击 Display 按钮，显示如图 6-37 所示的流线图。

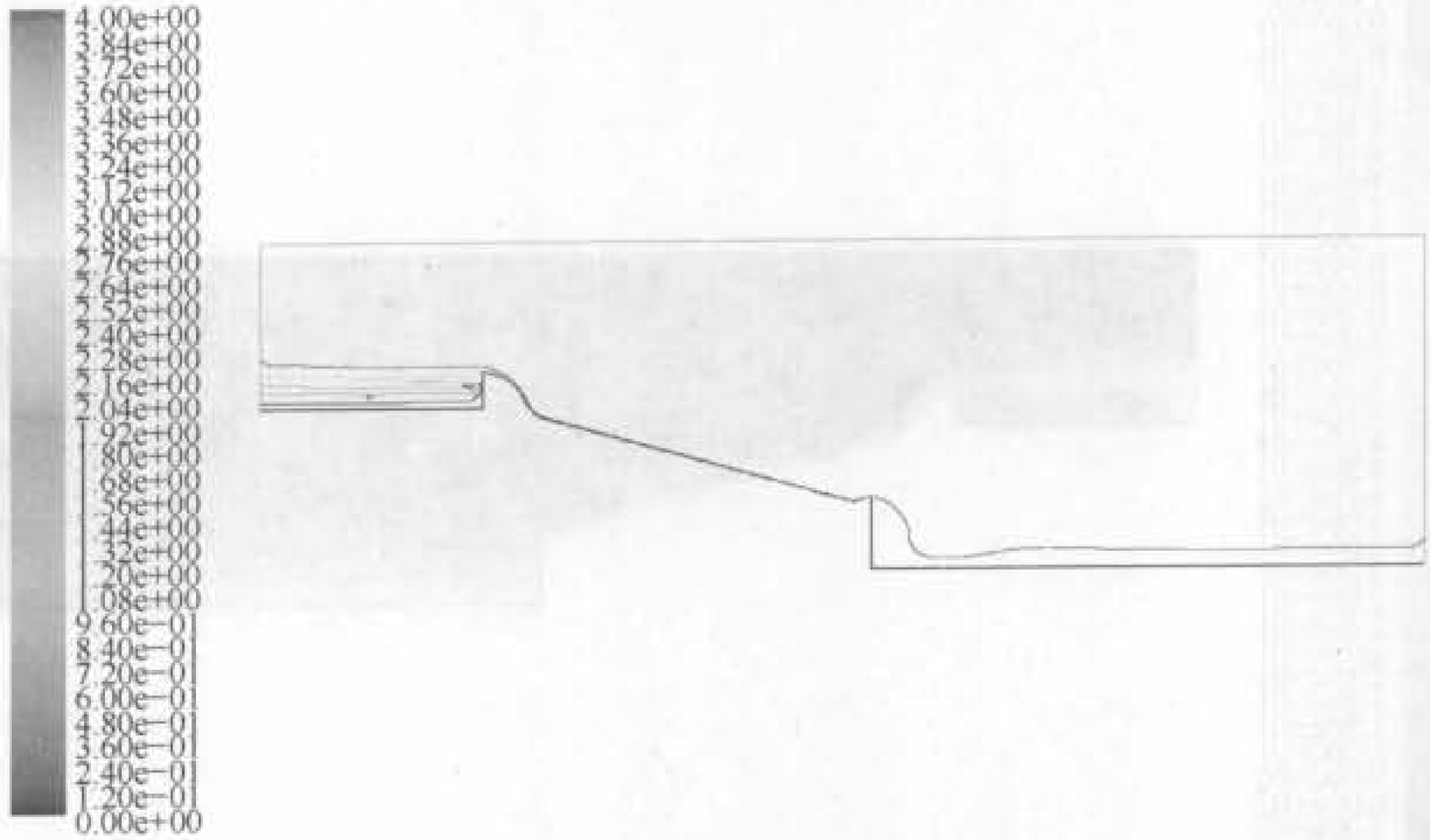


图 6-37 流线显示

4) XY 曲线显示

显示计算区域任意线段或任意面上的物理量的变化情况，这里在 sediments 面上创建一根线段。

选择 Surface→Line/Rake 命令，打开 Line/Rake Surface 对话框。输入 X0=540，X1=540，Y0=−150，Y1=150，单击 Create 按钮，创建名为 line-11(该名称根据自己的情况命名，这里是默认名称)的线段。

选择 Plot→XY Plot 命令，打开 Solution XY Plot 对话框。在 Y Axis Function 下拉列表框中选择 Phases，在 Phase 下拉列表框中选择 water，在 Surfaces 列表框中选择 line-11，单击 Plot 按钮，显示如图 6-38 所示的流线图。

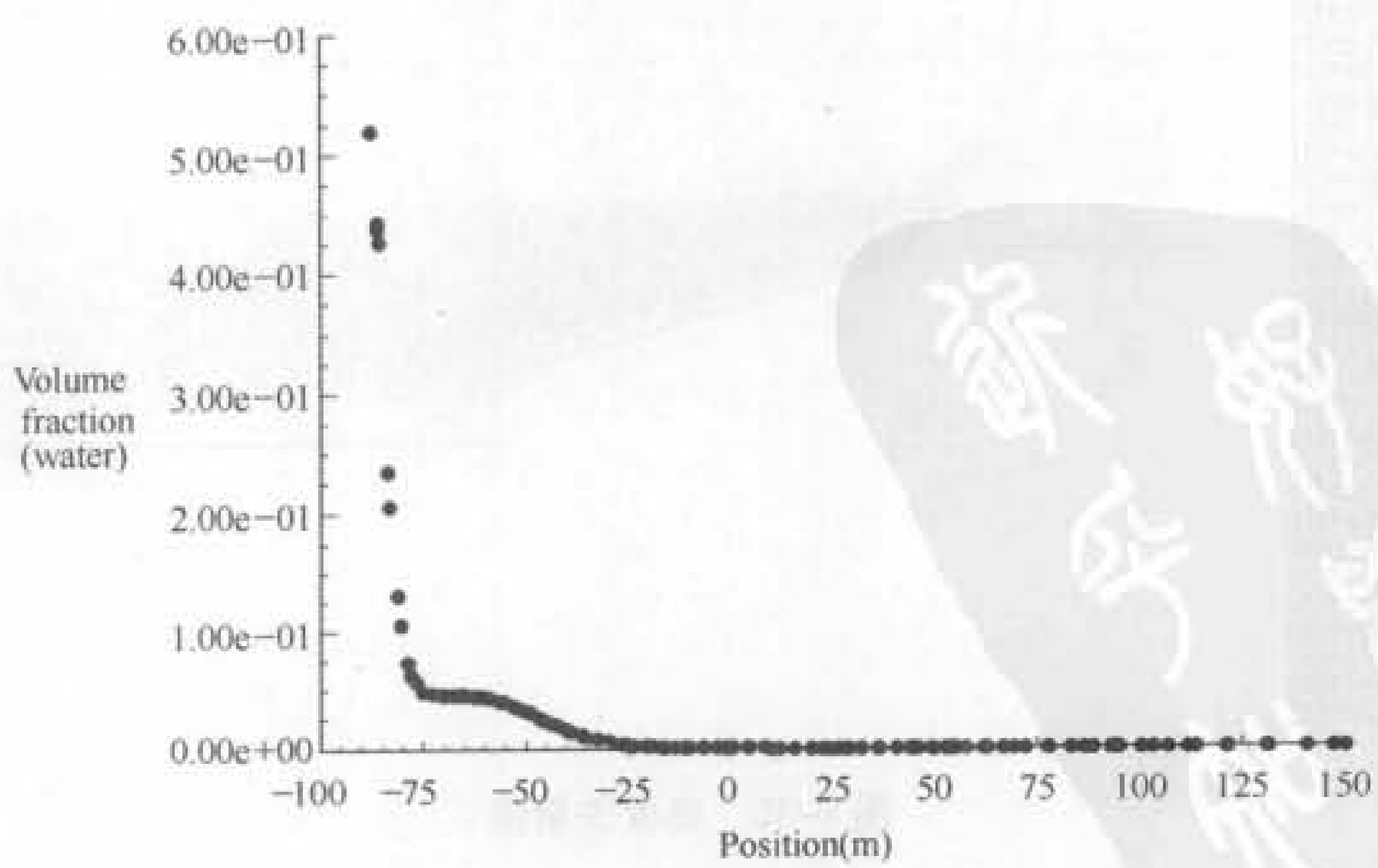


图 6-38 进口内侧一线段上 water 相分布曲线

第 7 章　动网格计算方法概述

在 Fluent 中，动网格模型可以用来模拟由于流域边界运动引起流域形状随时间变化的流动情况。这种流动情况既可以是一种指定的运动(比如，指定围绕物体重心随时间变化，以一定的线速度或角速度运动)，也可以是一个未确定的运动(比如这种运动的绕物体重心的线速度或角速度是由流域中固体上的受力平衡得出的)，下一时间步的运动情况是由当前时间步的计算结果确定的。各个时间步的体网格的更新是基于边界条件新的位置，由 Fluent 自动来完成的。

7.1　动网格计算模型

对于通量ϕ，在任一控制体V内，其边界是运动的，守恒方程的通式为

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{V}\rho\phi\mathrm{d}V+\int_{\partial V}\rho\phi(\vec{u}-\vec{u}_s)\cdot\mathrm{d}\vec{A}=\int_{\partial V}\Gamma\nabla\phi\cdot\mathrm{d}\vec{A}+\int_{V}S_{\phi}\mathrm{d}V \tag{7-1}$$

式中：ρ是液体的密度；$\vec{u}$是液体的速度矢量；$\vec{u_s}$是动网格的网格变形速度；Γ是扩散系数；S_ϕ是通量的源项ϕ；∂V代表控制体V的边界。

在方程(7-1)中，第一项可以用一阶向后差分形式表示为

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{V}\rho\phi\mathrm{d}V=\frac{(\rho\phi V)^{n+1}-(\rho\phi V)^{n}}{\Delta t} \tag{7-2}$$

式中，n和n+1 代表当前和紧接着的下一时间步的数值。第(n+1)步的体积V^{n+1}由式(7-3)计算得出：

$$V^{n+1}=V^{n}+\frac{\mathrm{d}V}{\mathrm{d}t}\Delta t \tag{7-3}$$

7.2　动网格更新方法

当运动条件定义在边界条件上时，Fluent 提供了三种动网格运动的方法来更新变形区域内的体网格：①基于弹性变形的网格调整；②动态的网格层变；③局部网格重构。

7.2.1　基于弹性变形的网格调整

对于三角形或四边形网格的流体区域，可以通过基于弹性变形的网格调整方法，根据边界节点上的已知位移来光滑调整流域内节点的位置。通过基于弹性变形的网格光滑调整更新体网格而没有改变网格之间的连通性。

在基于弹性变形的网格调整方法中，网格上任意两节点之间的连线被理想化成互相联结的弹簧。边界节点上给定的位移将产生一个与连接到这个节点所有弹簧位移成比例的力，

所以这样边界上节点的位移通过体网格就在流域中传播过去。从平衡角度来看，对每一个节点，连接到其上的所有弹簧的合力必为零。这个条件可以用以下的迭代方程表示：

$$\Delta\overrightarrow{x_i}^{m+1} = \frac{\sum_j^{n_i} k_{ij}\Delta\overrightarrow{x_j}^{m}}{\sum_j^{n_i} k_{ij}} \tag{7-4}$$

式中：$\Delta\overrightarrow{x_i}$ 是节点 i 的位移；n_i 是与节点 i 相邻的节点的数量；k_{ij} 是节点和与之相邻的节点之间的弹性常量系数，这里弹性系数可以定义为

$$k_{ij} = \frac{1}{\sqrt{|\overrightarrow{x_i} - \overrightarrow{x_j}|}} \tag{7-5}$$

由于边界上的位移是已知的，方程通过 Jacobi 矩阵对流域内部所有节点进行扫掠求解。在求解过程中，更新后的节点位置可以由式(7-6)表示：

$$\overrightarrow{x_i}^{n+1} = \overrightarrow{x_i}^{n} + \Delta\overrightarrow{x_i}^{k,\text{converged}} \tag{7-6}$$

如图 7-1 所示，圆柱体的一端运动，其网格动态弹性变形。

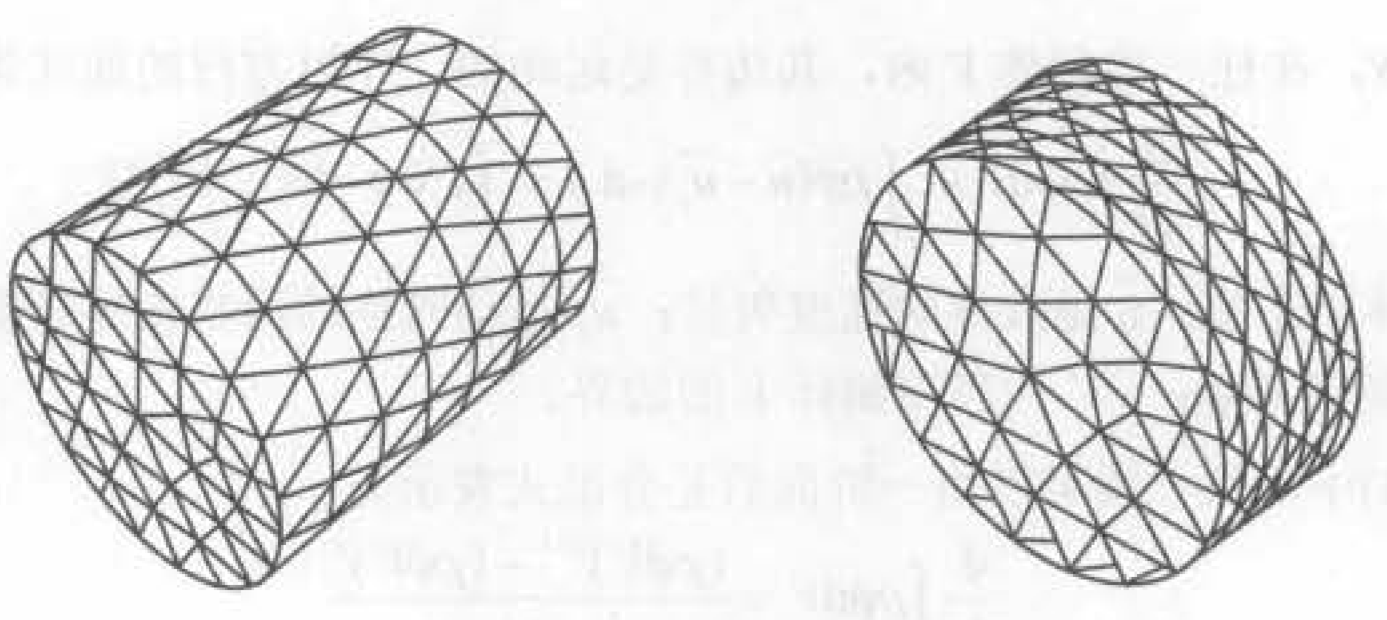


图 7-1 网格弹性变形前后

7.2.2 动态网格层变方法

在棱柱形(六面体或楔形)网格区域，动态层变方法可以根据与运动的物面邻近的网格层的高度来决定增加或减少网格的层数。在 Fluent 中，动网格模型可以指定一个理想的高度。邻近边界的网格单元层(图 7-2 中的 j 层)根据层 j 的单元层高度 h 来分裂出新的单元层或与邻近的层 i 合并成一个新层。

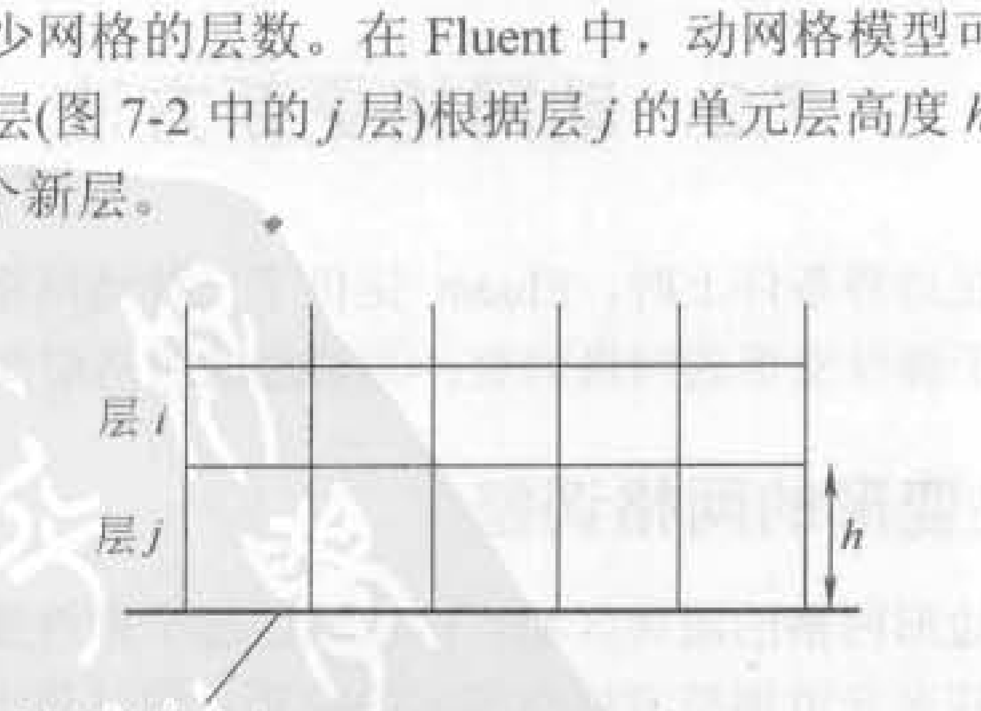


图 7-2 动态网格层

如果层 j 中单元体积是处于膨胀状态的，Fluent 允许它们膨胀到

$$(1+\alpha_h)h_{\text{ideal}} \tag{7-7}$$

式中：h_{ideal} 是理想单元高度，α_h 是全局单元层的分裂/合并因子。

当 $h>(1+\alpha_h)h_{\text{ideal}}$ 时，单元将根据预定义的高度条件进行分裂，也就是说，在层 j 中的单元将分裂成一个具有理想高度 h_{ideal} 的单元层和一个单元高度为 $(h-h_{\text{ideal}})$ 的单元层。如果层 j 中的单元体积是被压缩的，当压缩到 $h<\alpha_h h_{\text{ideal}}$ 时，这个被压缩的单元层将与邻近的单元层合并成一个新层；也就是说，在层 i 和层 j 中的单元将合并。

图 7-3 给出了内燃机邻近活塞壁的单元层的网格分解情况。

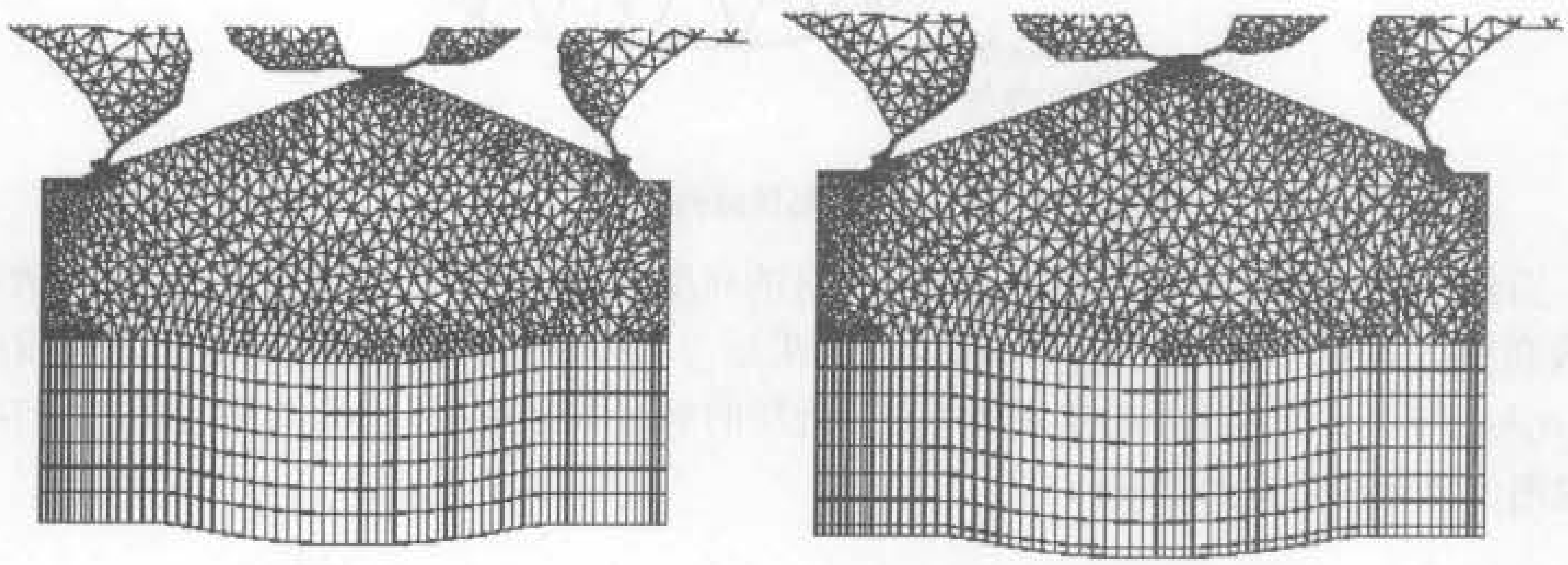


图 7-3　活塞壁的单元层的网格分解情况

7.2.3　局部网格重构方法

当边界位移相对局部单元尺寸较大时，单元质量将恶化或单元将退化，这样就导致下一时间步的求解收敛困难。为了避免这个问题，Fluent 把这些网格质量差的单元(单元尺寸太大或太小的单元，或者高度变形的单元)聚集成团，再对这个网格聚团区域进行重构。

Fluent 将检查流域内每一个单元的质量，并且对满足下列一个或多个条件的单元给予标注。

- 单元尺寸小于指定的单元尺寸最小值。
- 单元尺寸大于指定的单元尺寸最大值。
- 单元的畸变度大于指定的最大畸变度。

除了对体网格进行重构外，Fluent 还对变形边界上网格的三角形面或线性面随同体网格一起进行重构，这样的网格面如同图 7-4 给出了连续的圆柱形表面。而且，只有那些与运动节点相邻的网格面在考虑范围之内。这种网格重构技术类似于 7.2.2 节所描述的动态层变技术，除了它是应用于与运动网格节点相邻面网格的高度的网格运动。

这些与运动的物面邻近的网格单元面(图 7-4 中的 j 层)根据其高度 h 来确定分裂出新的单元层或与邻近的层(图 7-4 中的 i 层)合并成一个新层。

如果层 j 中单元面是处于膨胀状态的，Fluent 允许它们膨胀到

$$(1+\alpha_k)h_{\text{ideal}} \tag{7-8}$$

式中：h_{ideal} 是理想单元面高度，α_k 是高度系数。

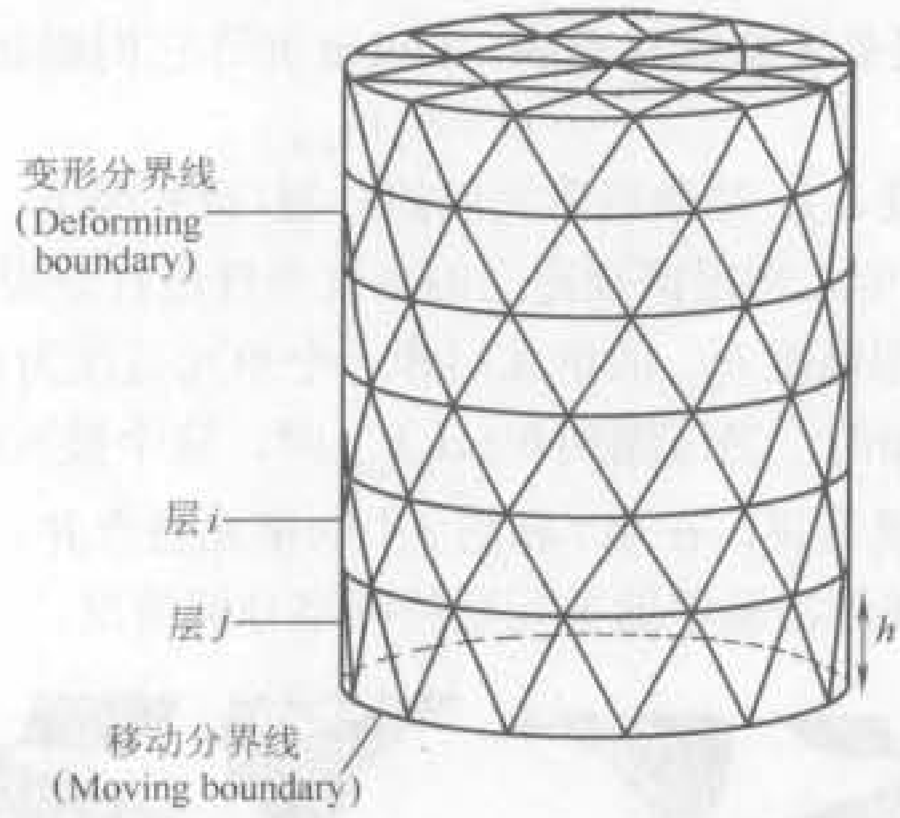


图 7-4 变形区域的重划分

当 $h > (1+\alpha_k)h_{\text{ideal}}$ 时，单元将根据预定义的高度条件进行分裂，这时，在层 i 中的单元面的高度正好是理想高度 h_{ideal} 。相反，如果层 j 中的单元体积是被压缩的，当压缩到 $h < \alpha_h h_{\text{ideal}}$ 时，这些被压缩的单元面将与邻近层的单元面合并成一个新的单元层。图 7-5 举例说明了单元面重构的过程。

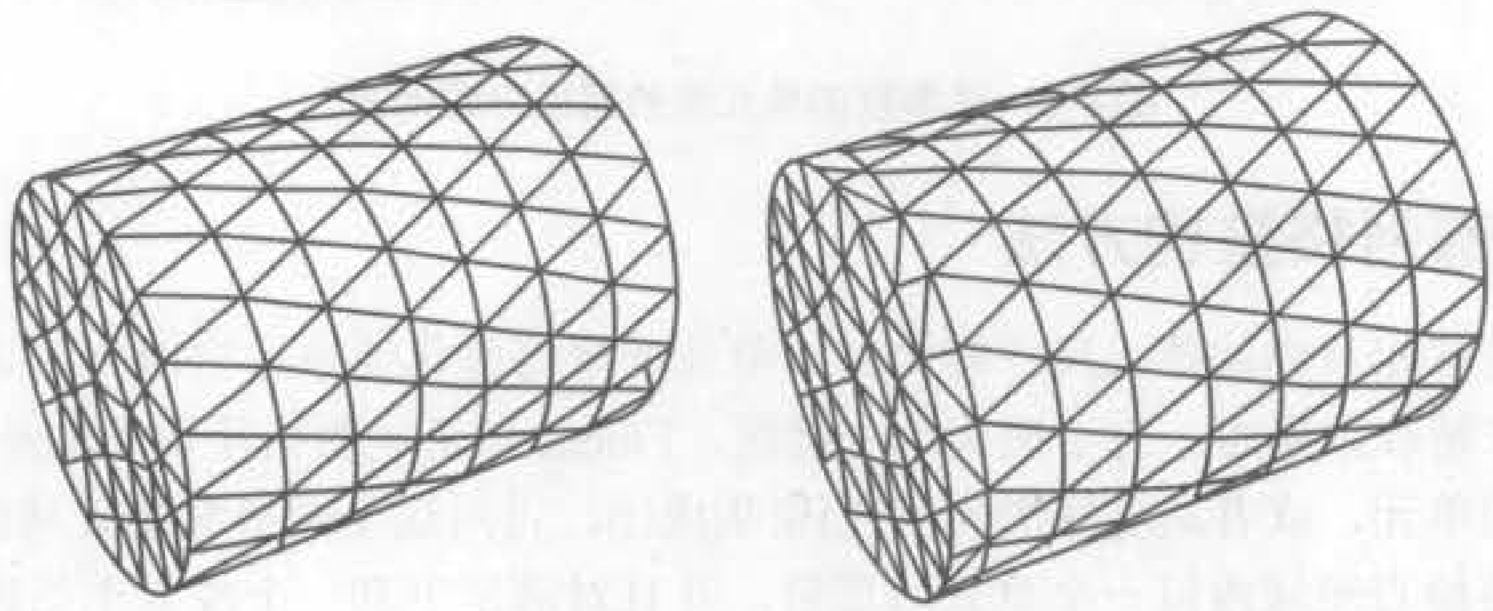


图 7-5 局部单元重构前后

7.3 Fluent 中动网格相关设置

7.3.1 启动动网格计算

在 Fluent 中，要进行动网格计算，需要激活动网格计算模块，并设置动网格总体计算参数及模式。

操作：选择 Define→Dynamic Mesh→Parameters 命令，打开如图 7-6 所示的 Dynamic Mesh Parameters(动网格参数)对话框，选中 Dynamic Mesh 复选框，会看到如图 7-7 所示的动网格详细设置对话框。

Models 选项组中有 Dynamic Mesh(动网格)、In-Cylinder(活塞内腔)、Six DOF Solver(六自由度求解器)等复选框，对于活塞腔内往复运动，需要选中 In-Cylinder 复选框；对于六自由度，如自由落体等，需选中 Six DOF Solver 复选框。一般情况下，只需选中 Dynamic Mesh

复选框。



图 7-6　动网格参数对话框

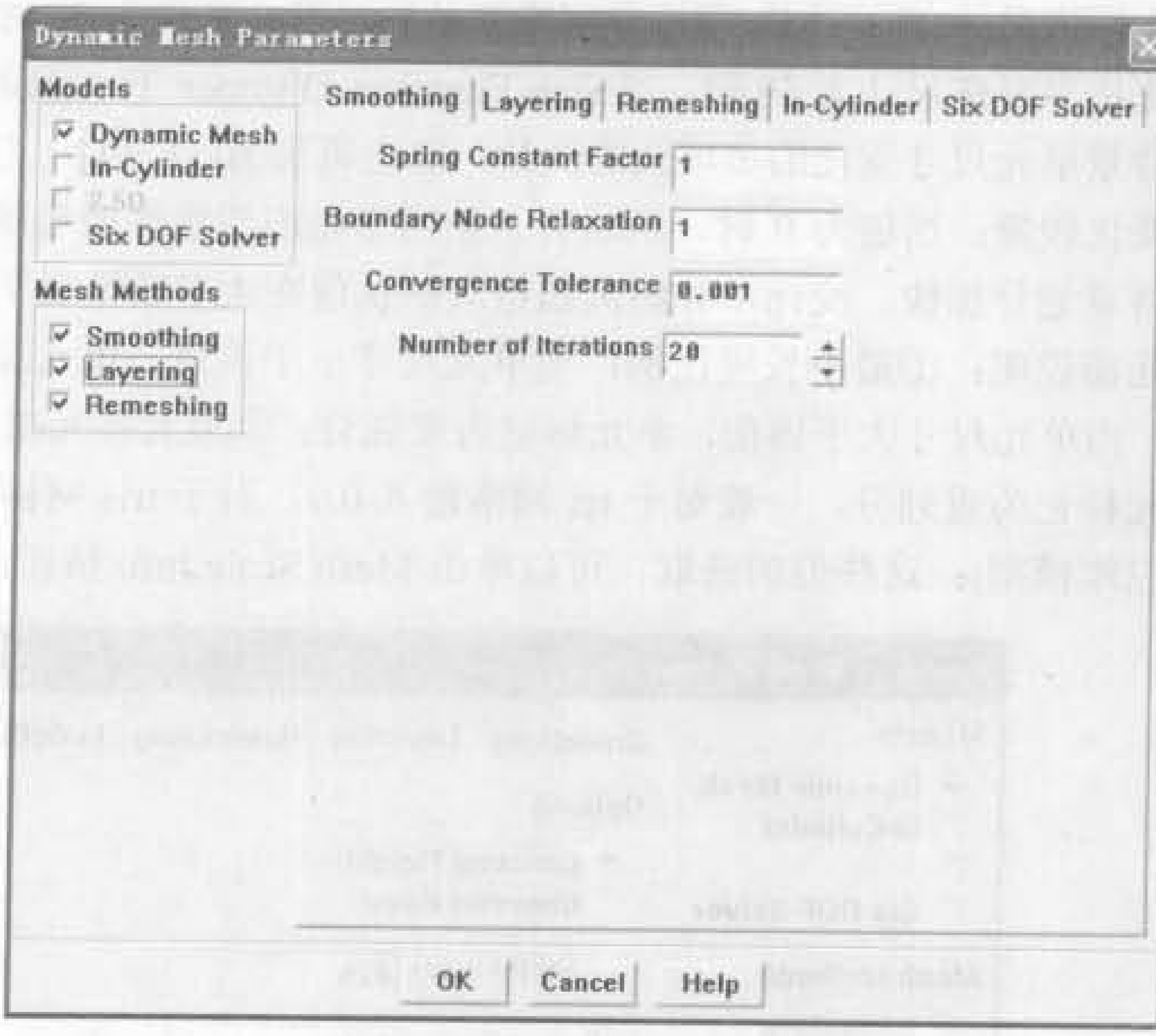


图 7-7　动网格参数设置对话框

Mesh Methods 选项组中有 Smoothing(网格光滑更新)、Layering(网格层变)、Remeshing(网格局部重划分)复选框，根据 7.2 节中所述内容进行选择。

当选择 Smoothing 网格更新方法时，需要网格光滑更新的参数，如图 7-7 所示，需要设置 Spring Constant Factor(弹性常数因子)、Boundary Node Relaxation(边界节点松弛)、Convergence Tolerance(收敛公差)、Number of Iterations(迭代数)。弹性常数因子给出弹簧的阻尼，取值范围为[0,1]。边界节点松弛，是针对边界节点的欠松弛，如果边界节点没有滑移变形，该值取 0，对于内部节点，采用默认值 1.0。收敛公差，一般在求解节点位置时采用。迭代数，是指最大的迭代次数。

当选择 Layering 网格更新方法时，相应的对话框如图 7-8 所示。在 Options 选项组中有 Constant Height (常数高度)和 Constant Ratio(常数变化率)两个单选按钮，常数高度用于统一的层高度的情况，而常数变化率用于有弯曲的区域，如圆柱体，层间的单元高度保持一个常数比率。另外还需输入 Split Factor(分裂因子)和 Collapse Factor(合并因子)，即对应于公式(7-7)中的 α_h(演化两个因子，分裂因子 α_s 和合并因子 α_c)，当 $h>(1+\alpha_s)h_{\text{ideal}}$ 时，单元将根据预定义的高度条件进行分裂；当压缩到 $h<\alpha_c h_{\text{ideal}}$ 时，这个被压缩的单元层将与邻近的单元层合并成一个新层。

当选择 Remeshing 网格重划分时，相应的对话框如图 7-9 所示。在 Options(选项)选项组中有 Sizing Function(尺寸函数)和 Must Improve Skewness(必须改善扭曲)、Face Remeshing(面重划分，仅对三维模型)三个复选框。当采用尺寸函数时，需要说明如下几点。①Size Function Resolution(尺寸函数求解)：决定用于估计相当于当前网格最小特征长度的尺寸分布背景仓的大小。②Sizing Function Variation Rates(尺寸函数的变化率)：反映背景网

格在尺寸方面(从边界到内部)怎么增加，取值范围为[-1, +inf]。当值为-0.5 时，说明内部的最小网格尺寸(单元边长)是边界网格尺寸的一半；当值为 50 时，意味着内部最大网格尺寸会比边界网格尺寸大 50 倍。③Size Function Distance Threshold(尺寸函数的距离阈值)：控制背景单元尺寸变化的多项式的次数，取值范围为[-1, +1]。当值较大时，边界区域网格尺寸变化较慢；当值为 0 时，从边界开始网格相应的线性增加或减少；当值为负数时，边界网格重划分加快。变化率和距离法拾取默认值在大多数情况下能取得较好的计算效果。另外还需说明：①最小长度比例：当单元尺寸小于该值，单元标记为重划分；②最大长度比例：当单元尺寸大于该值，单元标记为重划分；③最大单元扭曲：如果扭曲度超出这个值，单元标记为重划分，一般对于 tet 网格输入 0.9，对于 tris 网格输入 0.75；④最大面扭曲(仅对三维模型)：这些值的选取，可以单击 Mesh Scale Info 按钮，根据给出的值参考填入。

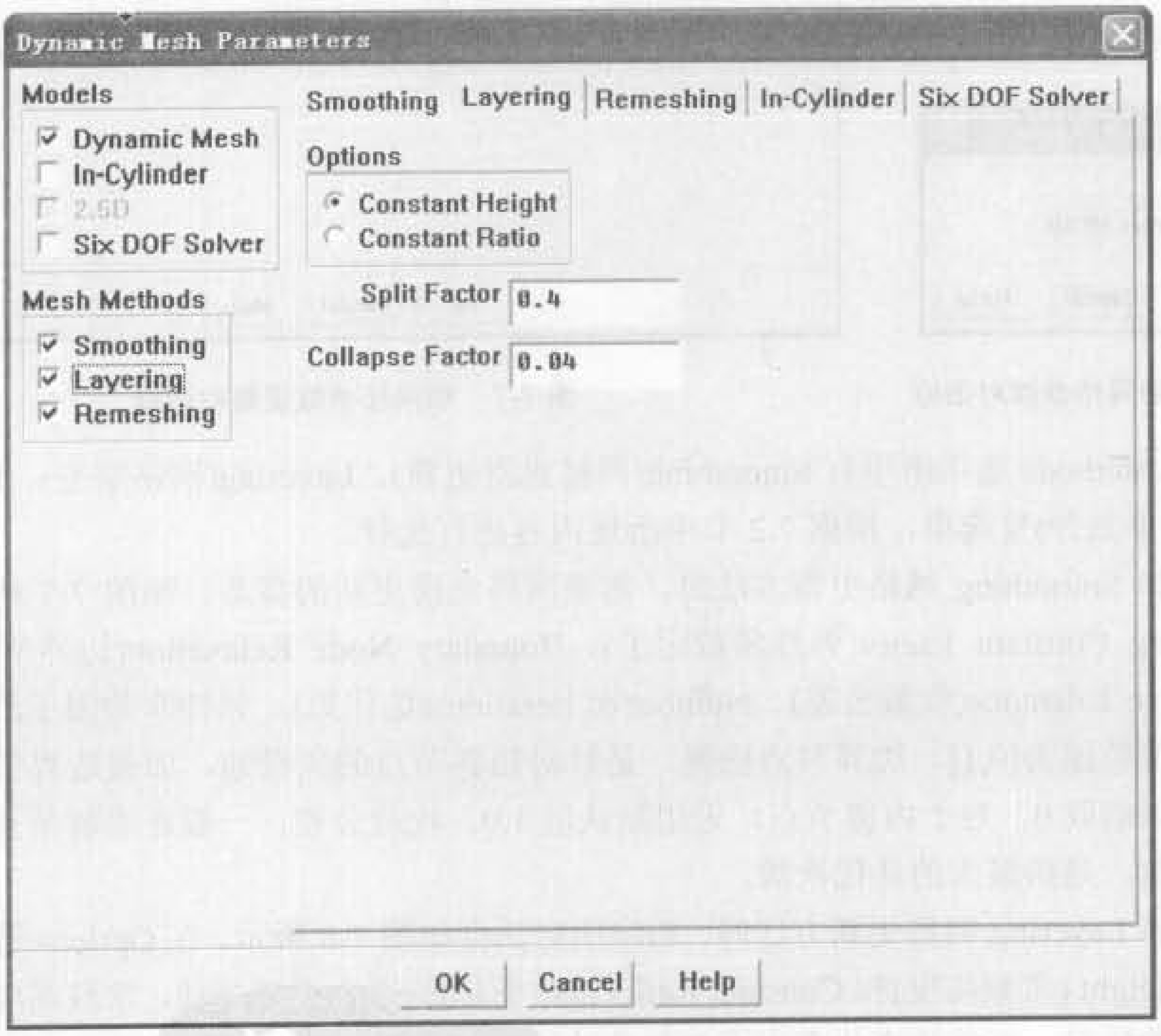


图 7-8 层网格更新参数设置对话框

当选中 In-Cylinder 时，相应的对话框如图 7-10 所示。需要输入：①Crank Shaft Speed(曲柄轴)；②Starting Crank Angle(曲柄轴的起始角度)；③Crank Period (曲柄周期)，对于四冲程发动机，是在 720° 的曲柄周期内完成一个循环，对于两冲程发动机，是在 360° 的曲柄周期内完成一个循环；④Piston Stroke(活塞冲程)；⑤Connecting Rod Length(连杆长度)；⑥Piston Stroke Cutoff(活塞冲程限位)；⑦Minimum Valve Lift(最小的阈提升值)。

当选中 Six DOF Solver 复选框时，相应的对话框如图 7-11 所示，主要是输入重力加速度的方向。

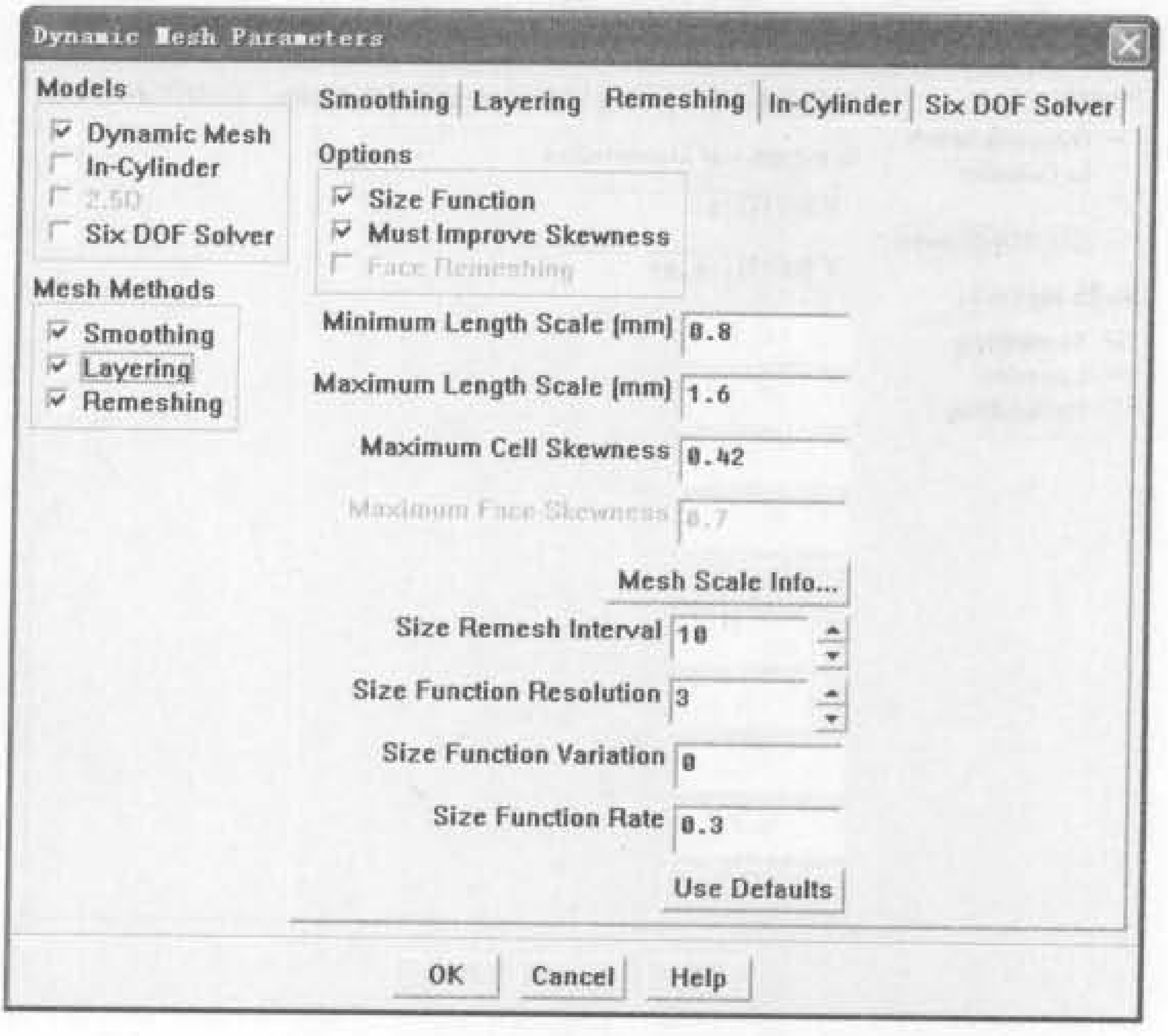


图 7-9 局部重划分网格更新参数设置对话框

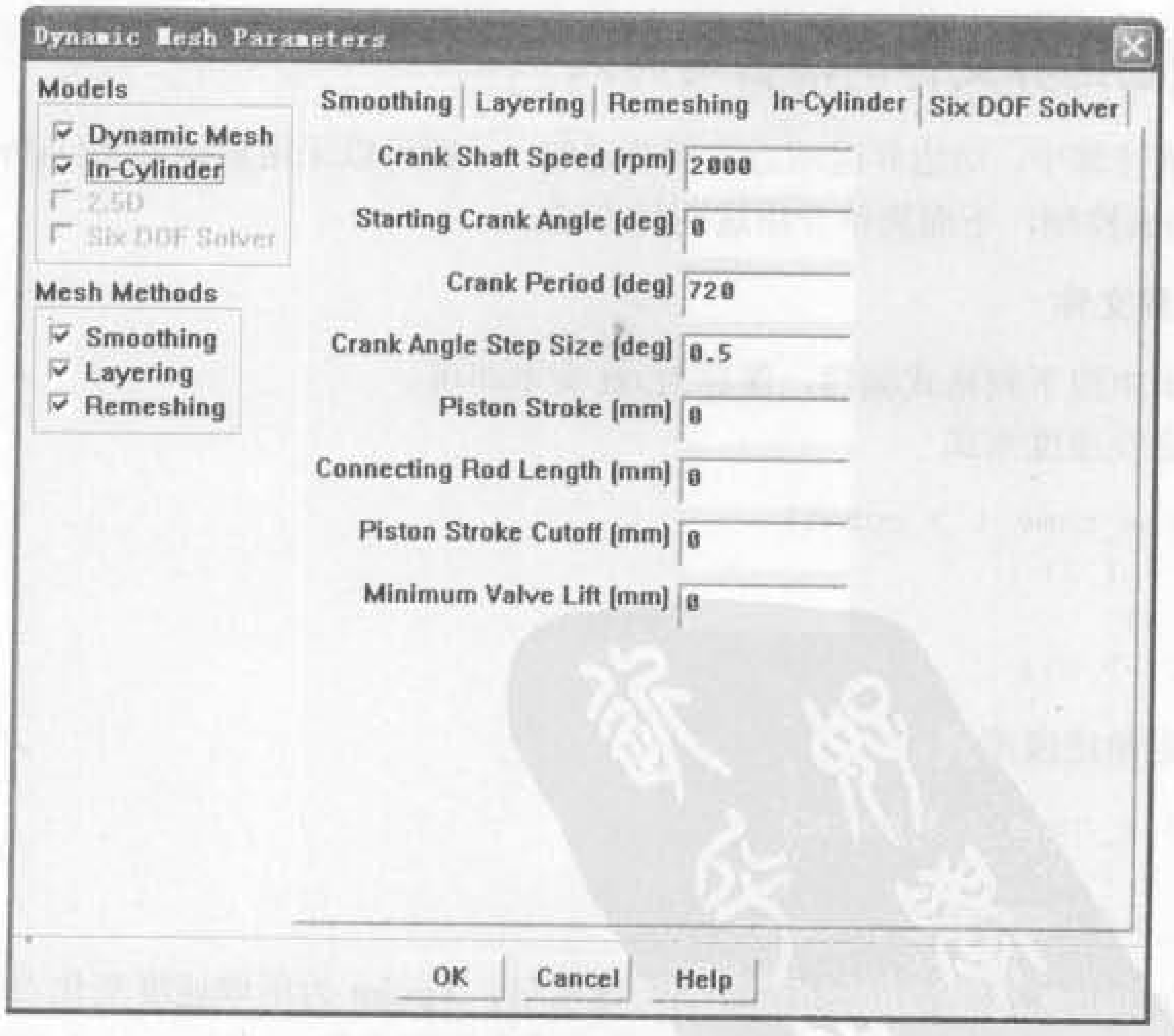


图 7-10 活塞内腔参数设置对话框

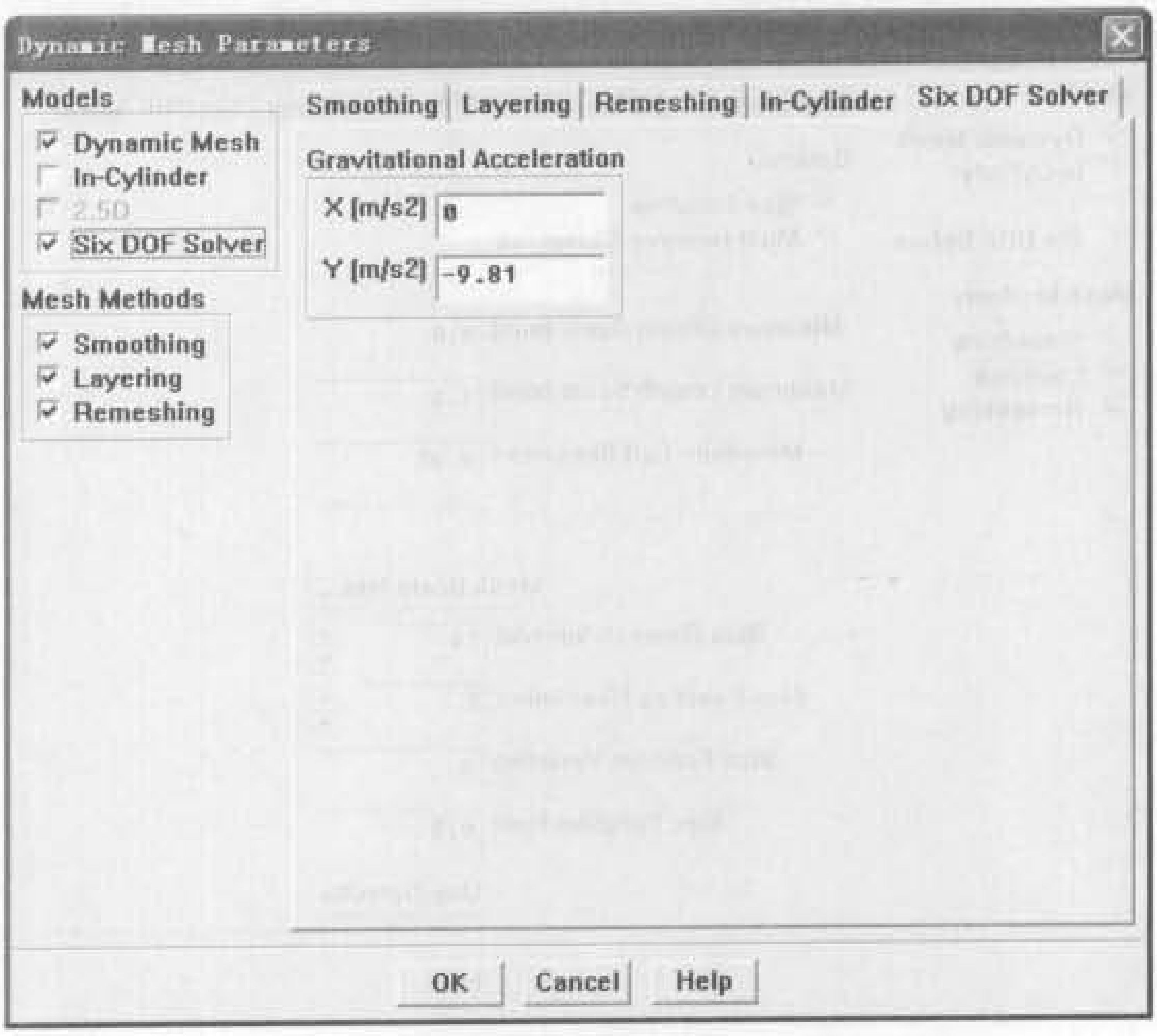


图 7-11　六自由度求解参数设置对话框

7.3.2　运动边界文件的准备与导入

在动网格计算中，动边界运动方式需要说明，一般可以采用动边界文件(Profile)和自定义函数(UDF)来控制，下面简单介绍这两种方法。

1. 动边界文件

在记事本中按下列格式编写，保存为.txt 文本即可。

(1) 给定线速度形式

```
((profile_name_1 3 point)
 (time 0 1 2)
(x 2 3 4)
 (v_y 0 -5 0))
```

(2) 给定角速度形式

```
((profile_name_2 3 point)
 (time 0 1 2)
(omega_x 2 3 4))
```

其中，profile_name_1 和 profile_name_2 为速度名；3 point 为所取速度变化点数，这里取 3 个点；time 后所述为所取点的时刻值；x 后所述为所取点的 x 坐标；v_y 后为所取点的 y 向速度；omega_x 后所述为所取点绕 x 轴的角速度。可以根据实际情况设定速度点数和各方向的速度等。

在 Fluent 中，当激活了动网格计算，选择 Define→Profile 命令，会出现如图 7-12 所示

的 Boundary Profiles 对话框。单击 Read 按钮，选择需要读入的 Profile 文件，就会在此对话框的 Profile 和 Fields 列表框中出现所设定的内容，单击 Close 按钮结束 Profile 的导入。

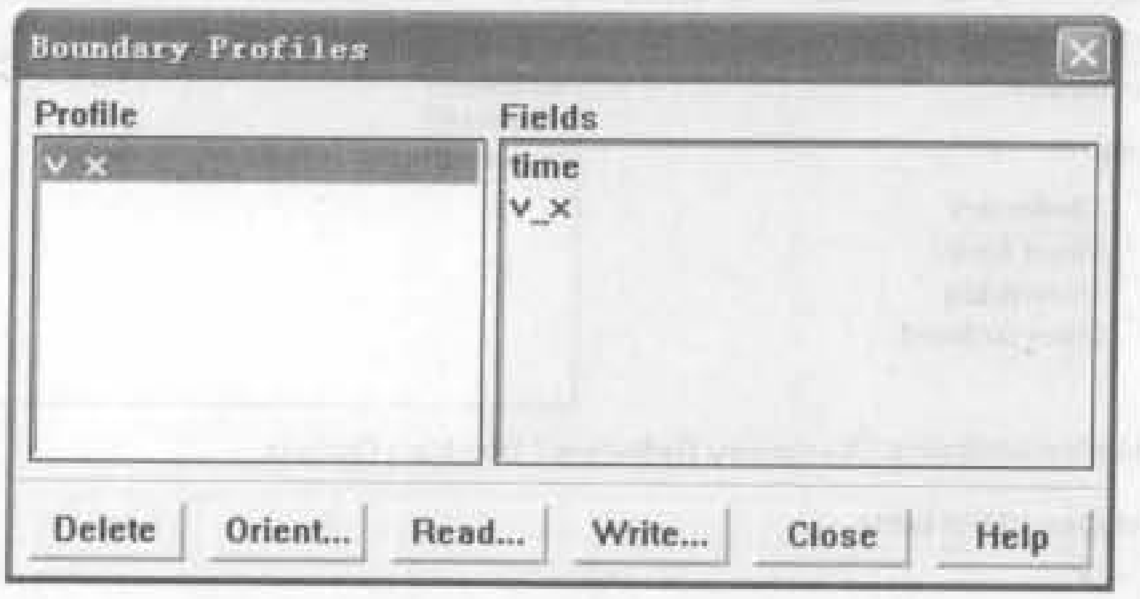


图 7-12　Profile 文件导入对话框

2. UDF 函数

对于比较复杂的动边界运动，需要采用 UDF 函数进行说明，Fluent 中专门设置了三个宏来描述动网格的运动，即 DEFINE_CG_MOTION 、DEFINE_GEOM 和 DEFINE_GRID_MOTION，当编写好这些函数，并编译成功后导入到 Fluent 中即可实现所需的运动，具体在第 8 章详细说明。

7.3.3　运动边界(动态区域)的相关设置

当激活动网格计算及输入运动边界文件后，需要所要求的运动边界类型和运动边界与运动条件相对应，因此通过 Define→Dynamic Mesh→Zones 命令激活动网格区域说明对话框(如图 7-13 所示)，进行具体参数输入。在 Zone Names 中选择所要设定运动的边界名称，在 Type 中选择运动边界类型：Stationary(静止)、Rigid Body(刚性体)、Deforming(变形)、User-Defined(用户定义)，对于不同类型，有不同的 Motion Attributes(运动属性)、Geometry Definition(几何定义)和 Meshing Options(网格选项)需要说明。然后单击 Create 按钮创建一个运动边界，该边界的名称就会出现在 Dynamic Zones(运动区域)列表框中。下面主要说明刚性运动边界和变形运动边界的设置。

对于刚性运动边界，选择 Rigid Body 单选按钮，需要说明运动属性和网格选项，如图 7-13 所示，在 Motion Attributes 选项卡中，在 Motion UDF/Profile 下拉列表框中选择合适的运动文件或函数，在 Center of Gravity Location 选项组中输入运动物体的重心，在 Center of Gravity Orientation 选项组中说明重心的方向。如图 7-14 所示，在 Meshing Options 选项卡中，输入单元高度(Cell Height)，对于 tet/tri 类型的局部网格重划分方式，这个高度是边界节点的距离；对于层网格重划分方式，该高度是理想高度，大致与初始网格高度相同。

对于变形边界，选中 Deforming 单选按钮，需要说明几何定义和网格选项，如图 7-15 所示，在 Geometry Definition 选项卡中，在 Definition 下拉列表框中选择合适的区域几何，有 faceted、plane、cylinder、user-defined 四个选项，对于 faceted，需要在 Feature Detection(特征检测)项目中决定是否包括特征及输入特征角度；对于 plane，如图 7-16 所示，需要说明 Point on Plane(面上的点)和 Plane Normal(面的法向)；对于 cylinder，如图 7-17 所示，需要说明 Cylinder Radius (圆柱体半径)、Cylinder Origin(圆柱体的原点)和 Cylinder Axis(圆柱

体轴)。

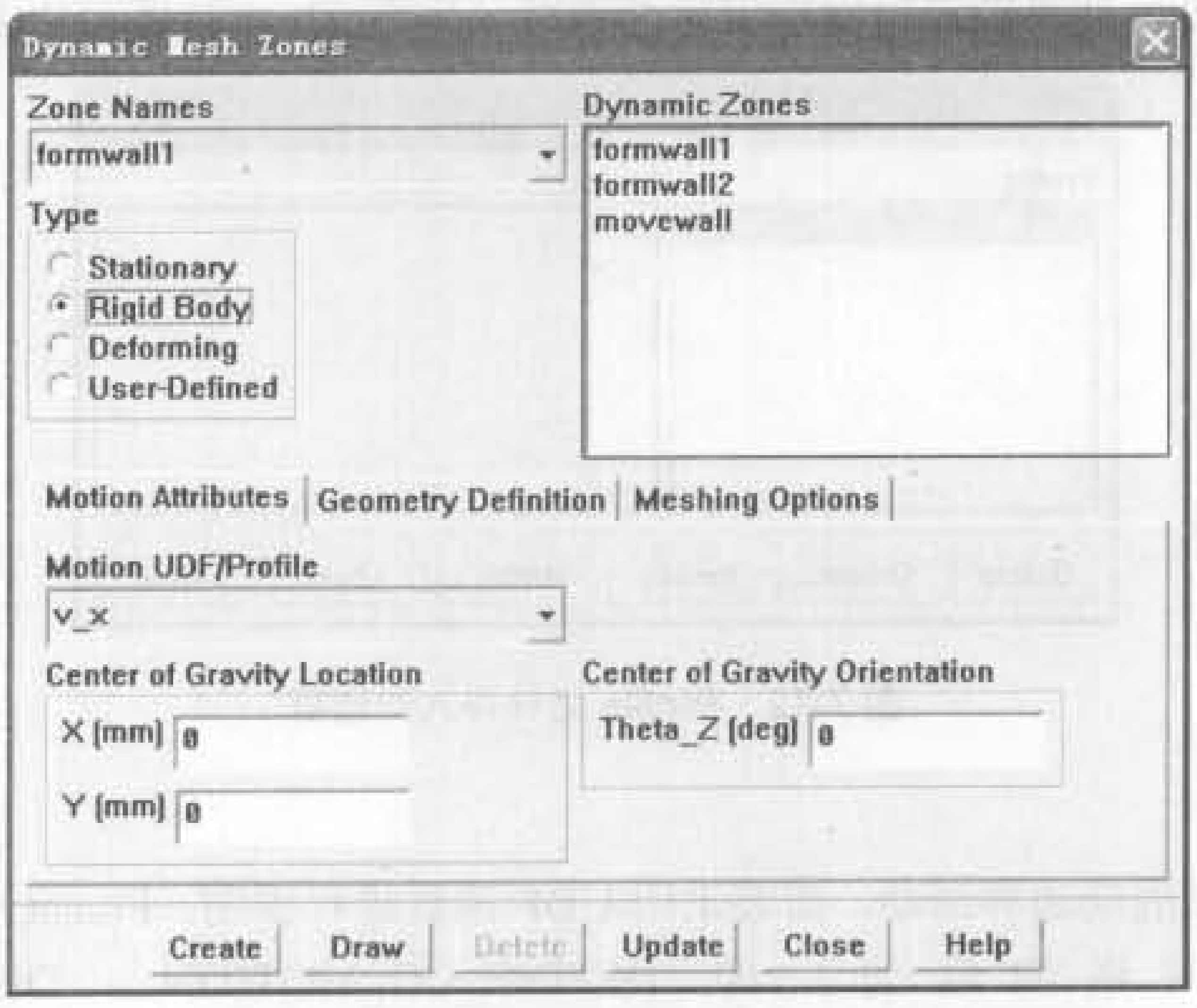


图 7-13　刚性体运动属性设置对话框

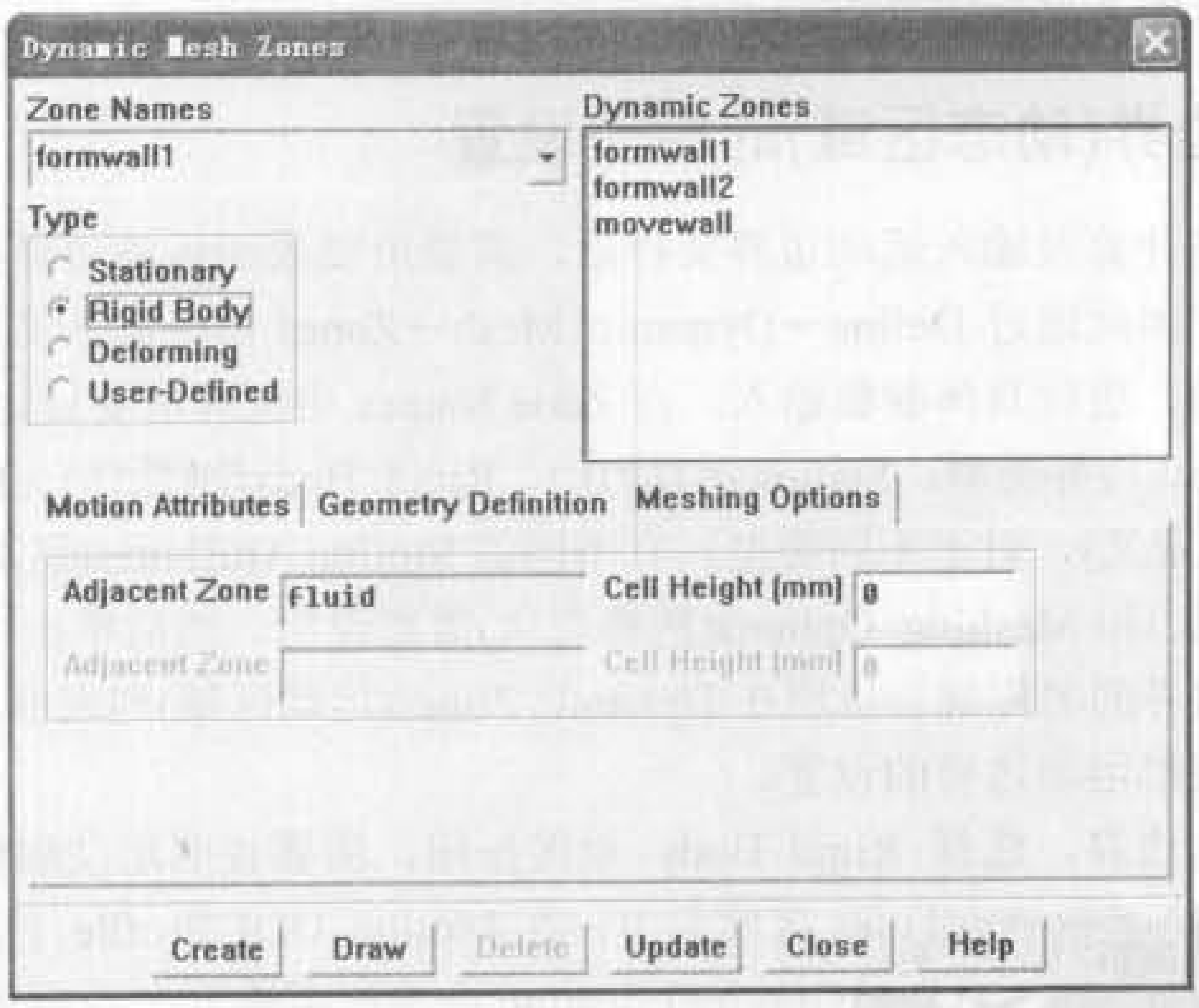


图 7-14　刚性体网格选项设置对话框

如图 7-18 所示，在 Meshing Options 选项卡中需要考虑的参数明显比刚性运动边界的多。在 Methods 选项组中，有可供选择的 Smoothing 和 Remeshing 两种网格更新方法复选框；在 Zone Parameters 选项组中，输入 Minimum Length Scale(最小长度比例)、Maximum Length Scale(最大长度比例)、Maximum Skewness(最大扭曲)，可以单击 Zone Scale Info 按钮得到这些参数的参考值；在 Smoothing Methods 选项组中说明光滑更新的方法，有 Spring 和 Laplace 两种，默认状态为 Spring；在 Remeshing Methods 选项组中说明重划分更新的方法，有 Region 和 Local 两种，默认状态下为 Region。

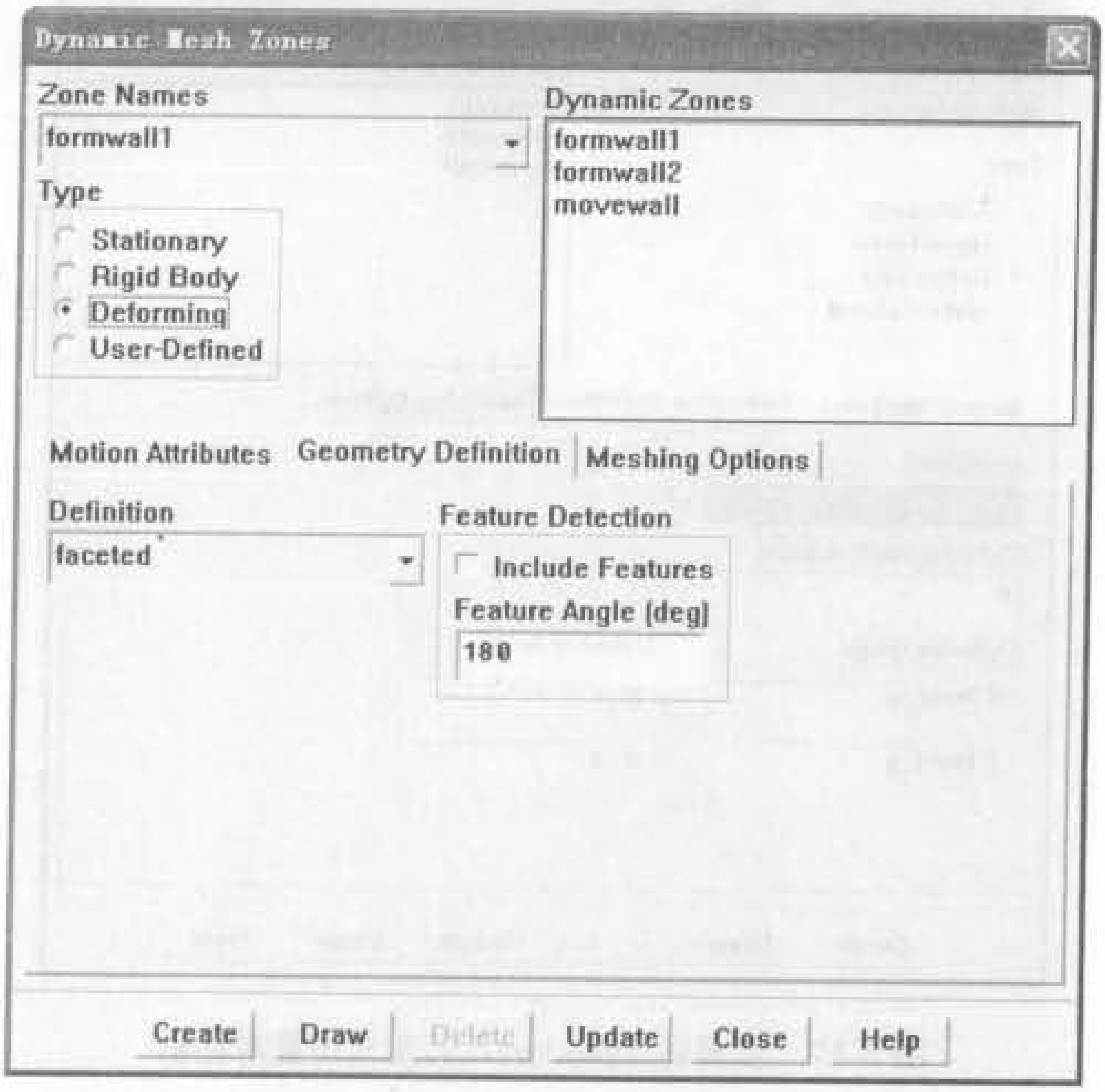


图 7-15　变形体几何定义设置对话框

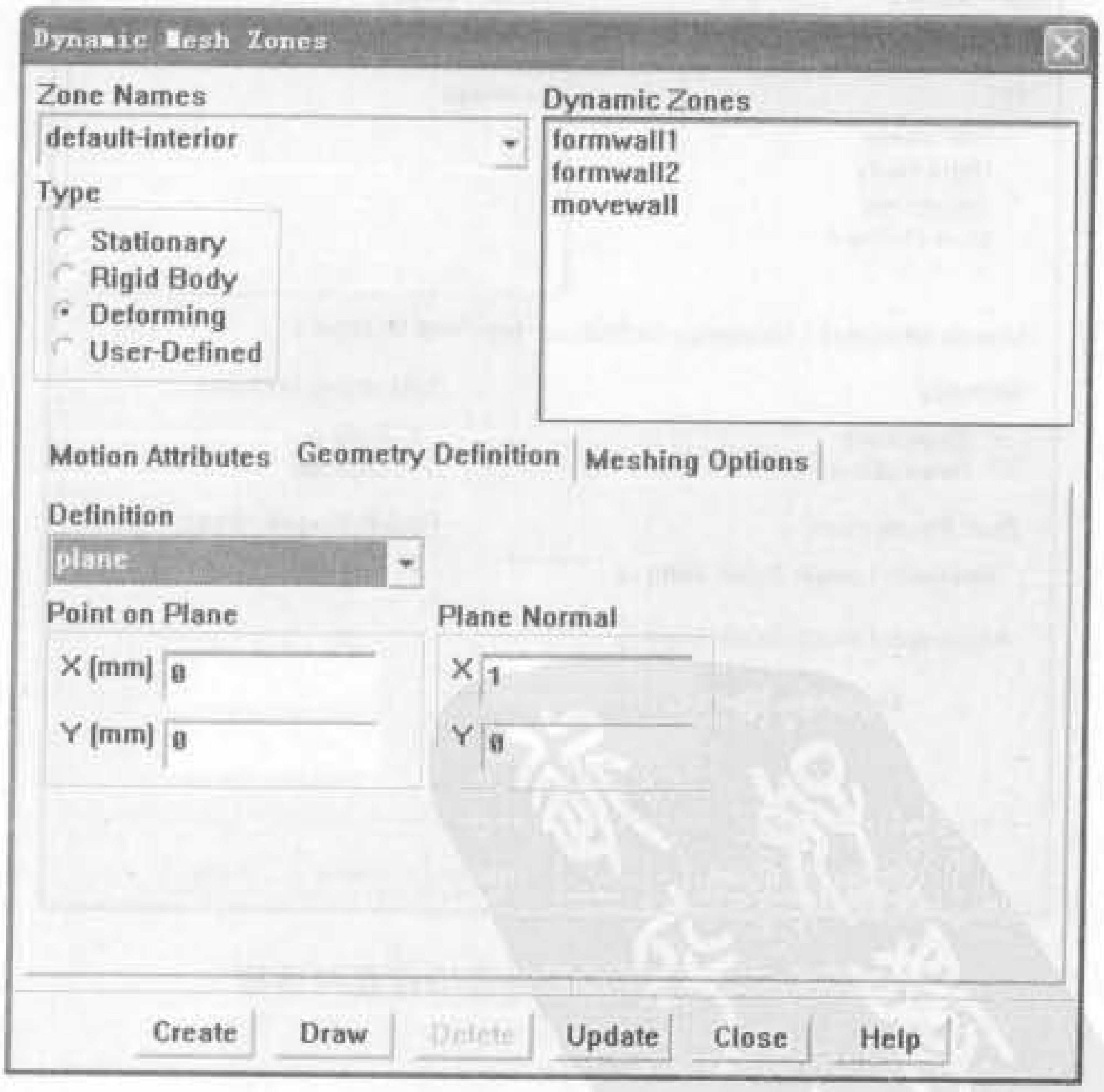


图 7-16　变形体(plane)几何定义设置对话框

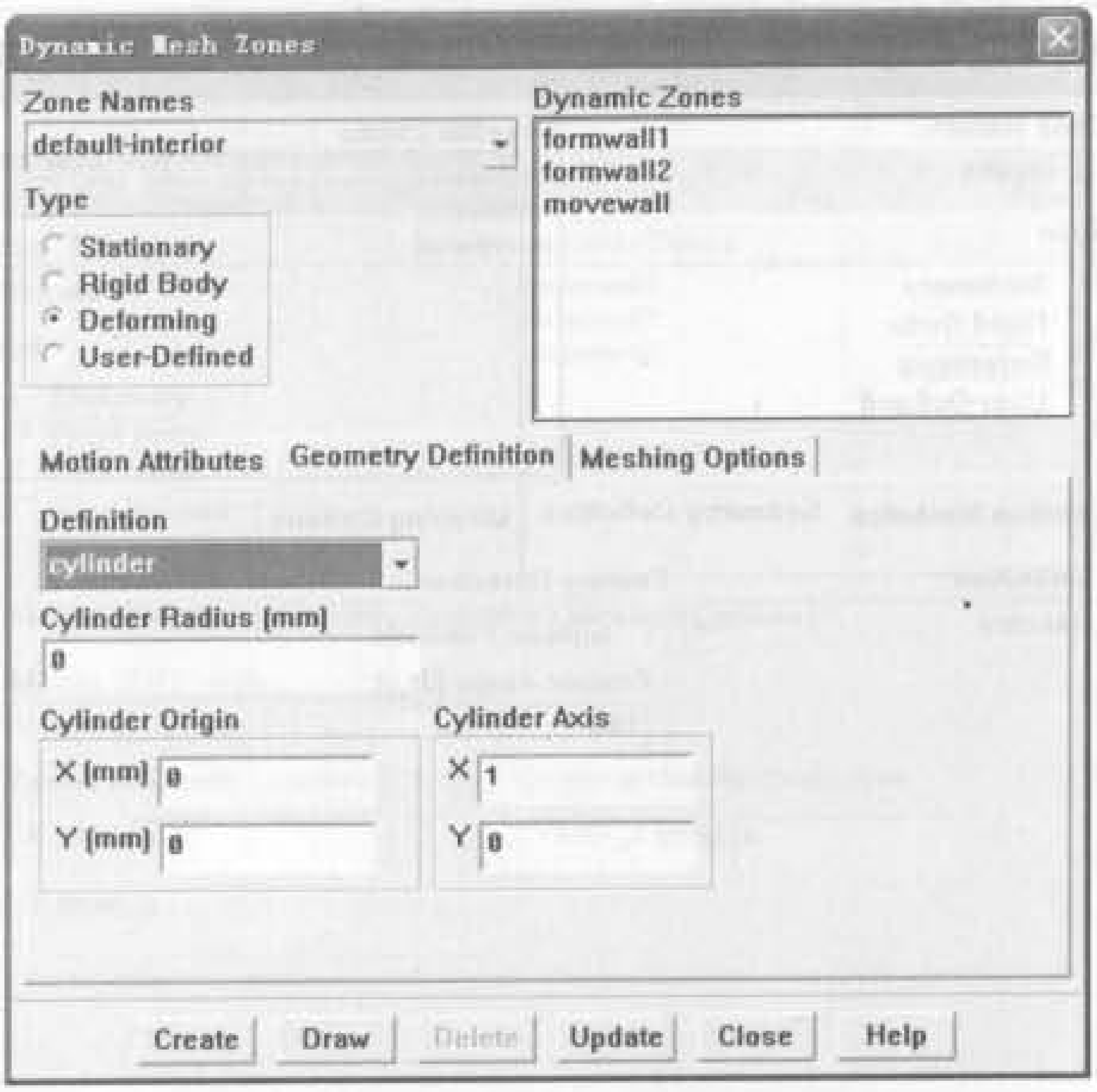


图 7-17　变形体(cylinder)几何定义设置对话框

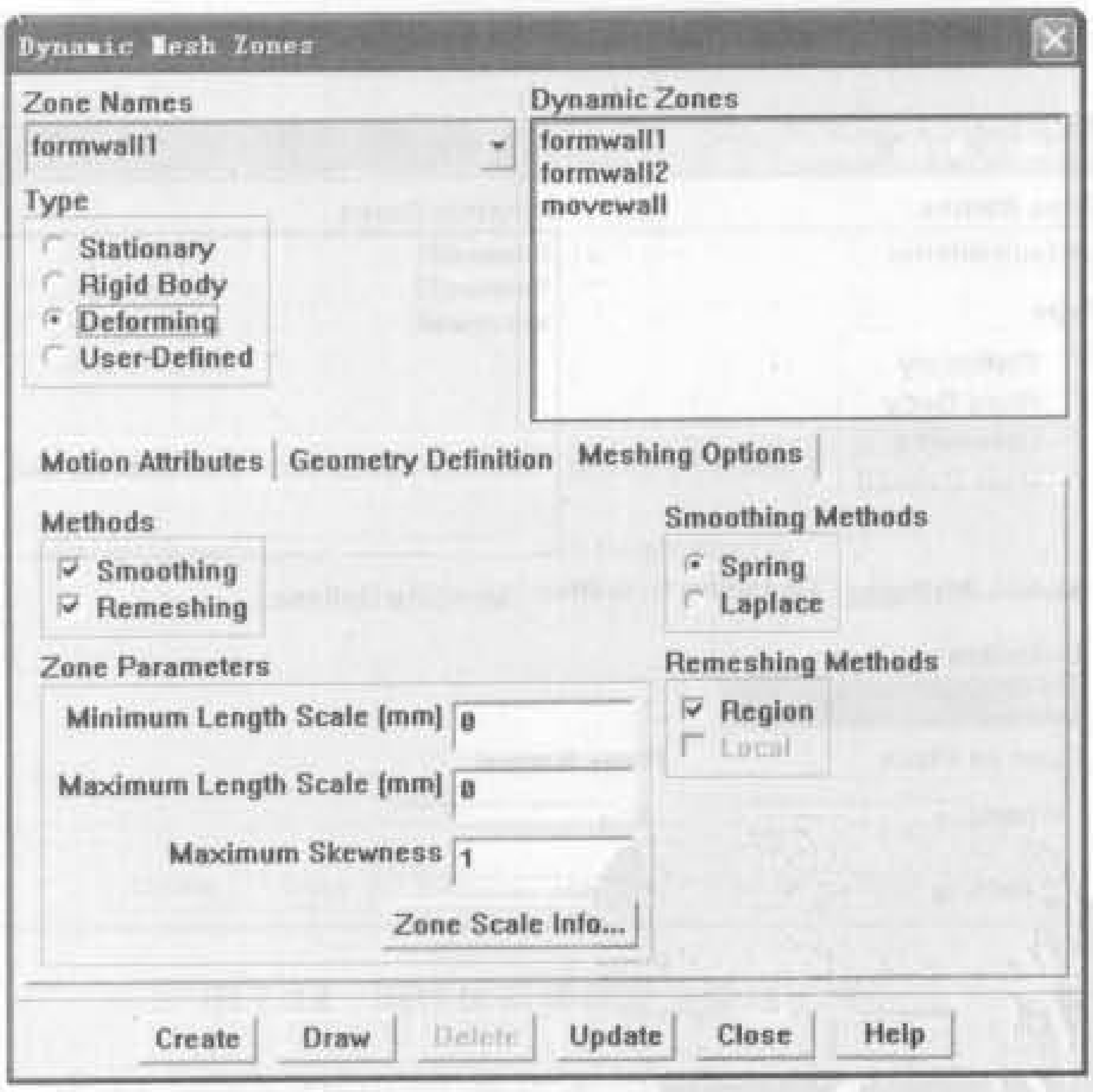


图 7-18　变形体网格选项设置对话框

- 面线索(face thread)：在其中分配了边界数据的面组。
- 节点线索(node thread)：节点组。
- 区域(domain)：由网格定义的所有节点、面和单元线索的组合。

8.1.3 Fluent 的数据类型

在编写 UDF 时，除了可以使用 C 语言数据类型外，还可以直接使用 Fluent 指定的与求解器数据相关的数据类型。常用的 Fluent 数据类型如下。

cell_t 是线索(thread)内单元标识符的数据类型，是一个识别给定线索内单元的整数索引。face_t 是线索内面标识符的数据类型，是一个识别给定线索内面的整数索引。

Thread 是单元或面的组合相关的数据容器。Node 是单元或面的拐角相关的数据容器。Domain 是 Fluent 中最高水平的数据结构，是一个与网格中所有节点、面和单元线索组合相关的数据容器。

8.2 UDF 中访问 Fluent 变量的宏

8.2.1 访问单元的宏

1. 访问单元流体变量的宏

在 Fluent 中可以用来访问单元上流体变量的宏在表 8-1 中列出，注意加了_G、_RG、_M1 和_M2 这些下标的单元格温度的宏，可以应用于表 8-1 中除了单元格压力(C_P)的所有求解器的变量中。这些下标分别表示的是矢量梯度、改造的矢量梯度、前一次的步长、和前两次的步长。而对于单元格压力，它的矢量梯度和相应的分量是使用 C_DP 得到的，而不是 C_P_G。每一个下标的描述和用法将在下面介绍。

表 8-1 在 mem.h 文件中的流体变量宏

名称(参数)	参数类型	返 回 值
C_T(c,t)	cell_t c, Thread *t	温度
C_T_G(c,t)	cell_t c, Thread *t	温度梯度矢量
C_T_G(c,t)[i]	cell_t c, Thread *t, int i	温度梯度矢量的分量
C_T_RG(c,t)	cell_t c, Thread *t	改造后的温度梯度矢量
C_T_RG(c,t)[i]	cell_t c, Thread *t, int i	改造后的温度梯度矢量的分量
C_T_M1(c,t)	cell_t c, Thread *t	温度的前一次步长
C_T_M2(c,t)	cell_t c, Thread *t	温度的前二次步长
C_P(c,t)	cell_t c, Thread *t	压力
C_DP(c,t)	cell_t c, Thread *t	压力梯度矢量
C_DP(c,t)[i]	cell_t c, Thread *t, int i	压力梯度矢量的分量
C_U(c,t)	cell_t c, Thread *t	u 方向的速度
C_V(c,t)	cell_t c, Thread *t	v 方向的速度

续表

名称(参数)	参数类型	返回值
C_W(c,t)	cell_t c, Thread *t	w 方向的速度
C_YI(c,t,i)	cell_t c, Thread *t, int i	物质质量分数
C_K(c,t)	cell_t c, Thread *t	湍流运动能
C_D(c,t)	cell_t c, Thread *t	湍流运动能的分散速率

(1) 可以在宏中加入下标_G 来得到梯度矢量和其分量。例如，C_T_G(c, t)就是返回单元格的温度梯度矢量。注意：只有当已经求解出包含这个变量的方程时才能得到梯度变量。例如，如果定义了一个关于能量的源程序，那么所编的 UDF 可以使用 C_ T _G 读写单元格的温度梯度，而不能使用 C_ U _G 读写 X 方向的速度分量。而且，如果建立了一个由使用者确定的方式转移方程，那么就不能得到一部分的梯度了。这是因为求解器不断地移走它不需要的数据。可以使用下面的方法来阻止存储器释放记忆：发出文本命令 save/set/expert，然后对计算机提出的“是否阻止暂时的求解器记忆释放”这一提问回答“是”，按照这种做法就可以保留所有的梯度数据，但是这种计算需要更多的内存。

(2) 在调用梯度矢量时把某一分量作为参数，这样就可以得到梯度分量了，参数 0 代表 X 方向的分量，1 代表 Y 方向的分量，2 代表 Z 方向的分量。例如，C_T_G(c, t)[0]就是返回温度梯度 X 方向的分量。注意：在表 8-1 中虽然只列出了温度梯度和其分量求解的宏，但是却可以扩展到除了压力以外的所有变量中去，对于压力只能按照表 8-1 中的方法使用 C_DP 来得到压力梯度和其分量。

(3) 和梯度一样的方式，可以通过加 RG 的下标在宏中得到梯度向量和其分量。通过使用恰当的整数作为参数来获得想要的矢量分量，参数 0 代表 X 方向的分量，1 代表 Y 方向的分量，2 代表 Z 方向的分量。当完成自己的插补计划时可以使用改造过的梯度。改造过的温度梯度和其分量在表 8-1 中列出了，同样可以推广到所有的变量。注意：改造过的梯度矢量和梯度矢量一样都只有在梯度方程被求解出来时才可以得到。

(4) 在表 8-1 里的宏中加入下标_M1 就可以得到前一次步长时间下($t-\Delta t$)的变量的值，得到的这些数据可以在瞬态的模拟中使用。例如，C_T_M1(c, t)，返回前一步时间下的单元格温度的值。若在表 8-1 里宏的后面加上下标_M2 就可以得到前两次步长下的时间($t-2\Delta t$)，这些数据可用于瞬态的模拟计算中。在表 8-1 中仅列出了温度的前一次步长的求法，也可以扩展到其他的变量中去。

2. 访问导数的宏

在表 8-2 中列出来的宏可以用于访问速度导数。

表 8-2 在 mem.h 文件中访问导数的宏

名称(参数)	参数类型	返回值
C_DUDX(c,t)	cell_t c, Thread *t	U 速度对 x 方向的导数
C_DUDX(c,t)	cell_t c, Thread *t	U 速度对 y 方向的导数
C_DUDZ(c,t)	cell_t c, Thread *t	U 速度对 z 方向的导数
C_DVDX(c,t)	cell_t c, Thread *t	V 速度对 x 方向的导数

续表

名称(参数)	参数类型	返回值
C_DVDY(c,t)	cell_t c, Thread *t	V速度对y方向的导数
C_DVDZ(c,t)	cell_t c, Thread *t	V速度对z方向的导数
C_DWDX(c,t)	cell_t c, Thread *t	W速度对x方向的导数
C_DWDY(c,t)	cell_t c, Thread *t	W速度对y方向的导数
C_DWDZ(c,t)	cell_t c, Thread *t	W速度对z方向的导数

3. 访问材料性质的宏

在表 8-3 中列出的宏可以用于存取材料的性质。

表 8-3 在 mem.h 文件中访问材料性质的宏

名称(参数)	参数类型	返回值
C_FMEAN(c,t)	cell_t c, Thread *t	第一次混合分数的平均值
C_FMEAN2(c,t)	cell_t c, Thread *t	第二次混合分数的平均值
C_FVAR(c,t)	cell_t c, Thread *t	第一次混合分数变量
C_FVAR2(c,t)	cell_t c, Thread *t	第二次混合分数变量
C_PREMIXC(c,t)	cell_t c, Thread *t	反应过程变量
C_LAM FLAME SPEED(c,t)	cell_t c, Thread *t	层流焰速度
C_CRITICAL STRAIN RATE(c,t)	cell_t c, Thread *t	临界应变速度
C_POLLUT(c,t,i)	cell_t c, Thread *t, int i	第i个污染物质的质量分数
C_R(c,t)	cell_t c, Thread *t	密度
C_MU L(c,t)	cell_t c, Thread *t	层流速度
C_MU T(c,t)	cell_t c, Thread *t	湍流速度
C_MU EFF(c,t)	cell_t c, Thread *t	有效粘度
C_K_L(c,t)	cell_t c, Thread *t	热传导系数
C_K_T(c,t)	cell_t c, Thread *t	湍流热传导系数
C_K_ EFF(c,t)	cell_t c, Thread *t	有效热传导系数
C_CP(c,t)	cell_t c, Thread *t	确定的热量
C_RGAS(c,t)	cell_t c, Thread *t	气体常数
C_DIFF L(c,t,i,j)	cell_t c, Thread *t, int i, int j	层流物质的扩散率
C_DIFF EFF(c,t,i)	cell_t c, Thread *t, int i	物质的有效扩散率
C_ABS COEFF(c,t)	cell_t c, Thread *t	吸附系数
C_SCAT COEFF(c,t)	cell_t c, Thread *t	扩散系数
C_NUT(c,t)	cell_t c, Thread *t	湍流速度

4. 访问用户自定义的单元标量和存储器的宏

在表 8-4 中列出的宏可以为单元格读写用户定义的标量和存储器。

表 8-4 在 mem.h 文件中的可以为单元格访问用户定义的标量和存储器的宏

名称(参数)	参数类型	返 回 值
C_UDSI(c,t,i)	cell_t c, thread *t, int i	用户定义的单元格标量
C_UDSI M(c,t,i)	cell_t c, thread *t, int i	前一次步长下用户定义的单元格标量
C_UDSI_DIFF(c,t,i)	cell_t c, thread *t, int i	用户定义的标量的单元格分散率
C_UDMI(c,t,i)	cell_t c, thread *t, int i	用户定义的单元格存储器

5. 访问雷诺兹压力模型的宏

在表 8-5 中列出了可以给雷诺兹压力模型读写变量的宏。

表 8-5 在 metric.h 文件中的可以给雷诺兹压力模型读写变量的宏

名称(参数)	参数类型	返 回 值
C_RUU(c,t)	cell_t c, Thread *t	*uu* 雷诺兹压力
C_RVV(c,t)	cell_t c, Thread *t	*vv* 雷诺兹压力
C_RWW(c,t)	cell_t c, Thread *t	*ww* 雷诺兹压力
C_RUV(c,t)	cell_t c, Thread *t	*uv* 雷诺兹压力
C_RVW(c,t)	cell_t c, Thread *t	*vw* 雷诺兹压力
C_RUW(c,t)	cell_t c, Thread *t	*uw* 雷诺兹压力

8.2.2 访问面的宏

1. 访问面流体变量的宏

在表 8-6 中列出的宏可以在边界面读写流体变量。注意，如果表面在边界上，那么流体的方向是由 F_FLUX 决定的点指向外围空间的。

表 8-6 在 mem.h 文件中可以在边界面读写流体变量的宏

名称(参数)	参数类型	返 回 值
F_R(f,t)	face_t f, Thread *t,	密度
F_P(f,t)	face_t f, Thread *t,	压力
F_U(f,t)	face_t f, Thread *t,	*u* 方向的速度
F_V(f,t)	face_t f, Thread *t,	*v* 方向的速度
F_W(f,t)	face_t f, Thread *t,	*w* 方向的速度
F_T(f,t)	face_t f, Thread *t,	温度
F_H(f,t)	face_t f, Thread *t,	焓
F_K(f t)	face_t f, Thread *t,	湍流运动能

续表

名称(参数)	参数类型	返 回 值
F_D(f,t)	face_t f, Thread *t,	湍流运动能的分散速率
F_YI(f,t,i)	face_t f, Thread *t, int i	物质的质量分数
F_FLUX(f,t)	face_t f, Thread *t	通过边界表面的质量流速

2. 读写用户定义的面标量和存储器的宏

在表 8-7 中列出了可以给表面读写用户定义的标量和存储器的宏。

表 8-7　在 mem.h 文件中的表面读写用户定义的标量和存储器的宏

名称(参数)	参数类型	返 回 值
F_UDSI(f,t,i)	face_t f, Thread *t, int i	用户确定的表面标量
F_UDMI(f,t,i)	face_t f, Thread *t, int i	用户定义的表面存储器

3. 访问混合面变量的宏

其余的表面变量宏在表 8-8 中列出，主要是访问混合面变量的宏。

表 8-8　在 mem.h 文件中的访问混合面的宏

名称(参数)	参数类型	返 回 值
F_C0(f,t)	face_t f, Thread *t	访问表面\0 边上的单元变量
F_C0_THREAD(f,t)	face_t f, Thread *t	访问表面\0 边上的单元线索
F_C1(f,t)	face_t f, Thread *t	访问表面\1 边上的单元变量
F_C1_ THREAD(f,t)	face_t f, Thread *t	访问表面\1 边上的单元线索

8.2.3　访问几何的宏

1. 节点和面的数量

在表 8-9 中列出的宏 C_NNODES 和 C_NFACES 返回相应的节点和面的整数值。对于一个给定的单元格， F_NNODES 返回与某个表面相关的节点的整数个数。

表 8-9　在 mem.h 文件中的节点和表面的宏

名称(参数)	参数类型	返 回 值
C_NNODES(c,t)	cell_t c,, Thread *t	一个单元格中的节点数
C_NFACES(c,t)	cell_t c,, Thread *t	一个单元格中的表面数
F_NNODES(f,t)	face_t f, Thread *t	一个表面中的节点数

2. 单元格和表面的重心

在表 8-10 中列出的宏可以用来获得一个单元格或是表面的真实的重心。C_CENTROID 找到单元格的重心的坐标，并把它的坐标存储在矩阵 X 中。F_CENTROID 找到表面的重心的坐标，并把它的坐标存储在矩阵 X 中。注意，矩阵 X 可以是一维、二维或者是三维的。

表 8-10 在 metric.h 文件中的变量重心宏

名称(参数)	参数类型	返回值
C_CENTROID(x,c,t)	real x[ND ND], cell_t c,, Thread * t	单元格重心的 x 值
F_CENTROID(x,f,t)	real x[ND ND], face_t f, Thread *t	表面中心的 x 值

3. 表面积

在表 8-11 中列出的宏 F_AREA 可以被用于返回一个实数的面积向量。对于内部的表面，标准的面积向量的方向是从单元格 C0 指向单元格 C1。标准的方向总是从边界面指向外(范围之外)的。

表 8-11 在 metric.h 文件中的表面积宏

名称(参数)	参数类型	返回值
F_AREA(A,f,t)	A[ND ND], face_t f, Thread *t	面积向量 A

4. 单元格体积

在表 8-12 中列出的宏可以用于获得二维、三维和轴对称的模型的单元格的真实体积。

表 8-12 在 mem.h 文件中的单元格体积宏

名称(参数)	参数类型	返回值
C_VOLUME(c,t)	cell_t c,, Thread *t	二维或是三维的单元格体积 单元格体积/2π 是轴对称模型的体积

8.2.4 访问节点的宏

表 8-13 和表 8-14 列出的宏返回单元格节点的实数直角坐标(在单元格的拐角)和相应的节点速度的分量。例如，在移动的网格模拟中节点速度是相对应的，每个变量的 Node *node 的参数定义了一个节点。这些宏的定义可以在相关的.h 文件中找到，如 mem.h 文件。

表 8-13 在 mem.h 文件中变量的节点坐标宏

名称(参数)	参数类型	返回值
NODE X(node)	Node *node	节点的 X 坐标
NODE Y(node)	Node *node	节点的 Y 坐标
NODE Z(node)	Node *node	节点的 Z 坐标

表 8-14 在 mem.h 文件中的节点速度变量宏

名称(参数)	参数类型	返回值
NODE GX(node)	Node *node	节点速度的 X 分量
NODE GY(node)	Node *node	节点速度的 Y 分量
NODE GZ(node)	Node *node	节点速度的 Z 分量

8.2.5 访问多相的宏

表 8-15 中列出的宏返回一个与整体多相节点相连的实数变量。这些变量的定义在 sg_mphase.h 文件中可以找到，这些包含在 udf.h 文件中。

表 8-15 在 sg_mphase.h 文件中的变量宏

名称(参数)	参数类型	返 回 值
C_VOF(c,pt[0])	cell_t c,, Thread **pt	主要相的体积分数
C_VOF(c,pt[n])	cell_t c,, Thread **pt	第 *n* 个辅助相的体积分数

8.3 UDF 实用工具宏

8.3.1 一般的循环宏

1. 查询控制区的单元线

当查询给定控制区的单元线索时，可以用 thread_loop_c。它包含单独的说明，后面是对控制区的单元线索所做的操作，正如下面显示的包含在{ }中。注意：thread_loop_c 在执行上和下面的 thread_loop_f 相似。

```
Domain *domain;
Thread *c_thread;
thread_loop_c(c_thread, domain) /*loops over all cell threads in domain*/
  {
  }
```

2. 查询控制区的面

当要查询给定控制区的面时，可以应用 thread_loop_f。它包含单独的说明，后面是对控制区的面单元所做的操作，正如下面显示的包含在{ }中。注意：thread_loop_f 在执行上和上面的 thread_loop_c 相似。

```
Thread *f_thread;
Domain *domain;
thread_loop_f(f_thread, domain)/* loops over all face threads in a domain*/
  {
  }
```

3. 查询单元线索中的单元

当要查询给定单元线索 c_thread 上所有的单元时，可以应用 begin_c_loop 和 end_c_loop。它包含 begin 和 end loop 的说明，完成对单元线索中单元所做的操作，定义包含在{ }中。当查找控制区单元线索的单元时，应用的 loop 全嵌套在 thread_loop_c 中。

```
cell_t c;
```

```
Thread *c_thread;
begin_c_loop(c, c_thread) /* loops over cells in a cell thread */
  {
  }
end_c_loop(c, c_thread).
```

4. 查询面线索中的面

当要查找给定面线索 f_thread 的所有的面时，可以用 begin_f_loop 和 end_f_loop。它包含 begin 和 end loop 的说明，完成对面线索中面单元所做的操作，定义包含在{ }中。当查找控制区面线索的所有面时，应用的 loop 全嵌套在 thread_loop_f 中。

```
face_t f;
Thread *f_thread;
begin_f_loop(f, f_thread) /* loops over faces in a face thread */
  {
  }
end_f_loop(f, f_thread)
```

5. 查询单元中的面

下面函数用以查询给定单元中所有面。它包含单独的查询说明，后面是所做的操作，定义包含在{}中。

```
face_t f;
Thread *tf;
int n;
c_face_loop(c, t, n) /* loops over all faces on a cell */
 {
 …
 f = C_FACE(c,t,n);
 tf = C_FACE_THREAD(c,t,n);
 …
 }
```

这里，n 是当前面的索引号。当前面的索引号用在 C_FACE 宏中以获得所有面的数量。例如，f = C_FACE(c,t,n)。

另一个在 c_face_loop 中有用的宏是 C_FACE_THREAD。这个宏用于合并两个面线索，例如，tf = C_FACE_THREAD(c,t,n)。

6. 查询单元节点

c_node_loop 函数用以查询给定单元中所有节点。它包含单独的查询说明，后面是所做的操作，定义包含在{}中。

```
cell_t c;
Thread *t;
int n;
c_node_loop(c, t, n)
  {
  …
```

```
node = C_NODE(c,t,n);
…
}
```

这里，n 是当前节点的索引号。当前节点的索引号用在 C_NODE 宏中以获得所有节点的数量。例如，node = C_NODE(c,t,n)。

8.3.2 查询多相组分的宏

1. 查询混合物中相的控制区

sub_domain_loop 宏用于查询混合物控制区的所有相的子区。这个宏查询并在混合物控制区,给每个相区定义指针以及相关的 phase_domain_index。控制区需要指针，在每个相中都有权访问部分数据。注意：sub_domain_loop 宏在执行中和下面的 sub_thread_loop 宏是相似的。

```
int phase_domain_index;    */ index of subdomain pointers */
Domain *mixture_domain;
Domain *subdomain;
sub_domain_loop(subdomain, mixture_domain, phase_domain_index)
```

sub_domain_loop 的变量是 subdomain、mixture_domain 和 phase_domain_index。

subdomain 是 phase-level domain 的指针，mixture_domain 是 mixture-level domain 的指针。当想用 DEFINE 宏时，mixture_domain(包含控制区变量，如 DEFINE_ADJUST)通过 Fluent 求解器自动传递给 UDF，混合物就和 UDF 相关了。如果 mixture_domain 没有显式地传递给 UDF，应用另外一个宏来恢复它，如在 sub_domain_loop 前使用 Get_Domain(1)。

phase_domain_index 是子区指针所引号，phase_domain_index 指向初始相时其索引号为 0，混合物中其他相依次加 1。注意：subdomain 和 phase_domain_index 是在 sub_domain_loop 宏定义中初始化的。

下面被集成在 UDF 中的语句在求解过程中补充说明一个相的体积分数，它在求解过程的开始执行。这个函数建立一个中心在点(0.5, 0.5, 0.5)，半径为 0.25 的球形体。第二个相的体积分数 1 被补充说明到球形体内的单元中，但是第二个相在其他单元中的体积分数为 0。

```
#include "udf.h"
/* domain pointer that is passed by INIT function is mixture domain */
DEFINE_INIT(my_init_function, mixture_domain)
{
  int phase_domain_index;
  cell_t cell;
  Thread *cell_thread;
  Domain *subdomain;
  real xc[ND_ND];
   /* loop over all subdomains (phases) in the superdomain (mixture) */
  sub_domain_loop(subdomain, mixture_domain, phase_domain_index)
  {
     /* loop if secondary phase */
     if (DOMAIN_ID(subdomain) == 2)
```

```
    /* loop over all cell threads in the secondary phase domain */
    thread_loop_c (cell_thread,subdomain)
    {
        /* loop over all cells in secondary phase cell threads */
        begin_c_loop_all (cell,cell_thread)
        {
            C_CENTROID(xc,cell,cell_thread);
            if (sqrt(ND_SUM(pow(xc[0] - 0.5,2.),
                            pow(xc[1] - 0.5,2.),
                            pow(xc[2] - 0.5,2.))) < 0.25)
                    /* set volume fraction to 1 for centroid */
                C_VOF(cell,cell_thread) = 1.;
            else
                /* otherwise initialize to zero */
                C_VOF(cell,cell_thread) = 0.;
        }
        end_c_loop_all (cell,cell_thread)
    }
  }
}
```

2. 查询混合物的相线索

sub_thread_loop 宏查询所有与混合物等值线索相同的等值线。这个宏查找并返回每个子区和相关 phase_domain_index 的指针。如果 subthread 指针与进口区域相关，那么这个宏将提供给进口区域每个相面线索指针。

```
int phase_domain_index;
Thread *subthread;
Thread *mixture_thread;
sub_thread_loop(subthread, mixture_thread, phase_domain_index)
```

sub_thread_loop 的自变量是 subthread、mixture_thread 和 phase_domain_index。subthread 是相线索的指针，mixture_thread 是 mixture-level thread 的指针。当用 DEFINE 宏(包含一个线索自变量)时，通过 Fluent 的求解器 mixture_thread 自动传递给 UDF，UDF 就和混合物相关了。如果 mixture_thread 没有显式地传递给 UDF，需要在调用 sub_thread_loop 之前调用工具宏恢复它。phase_domain_index 是子区指针索引号，可以用宏 PHASE_DOMAIN_INDEX 恢复。初始相的索引号为 0，混合物中其他相依次加 1。注意：subthread 和 phase_domain_index 在 sub_thread_loop 宏定义中被初始化。

3. 查询混合物中所有单元的线索

mp_thread_loop_c 宏查询混合物控制区所有单元的线索，提供了与混合物等值线有关的相等值线的指针。当应用到混合物控制区时这和 thread_loop_c 宏几乎是等价的，其区别是：除了查询每个单元线索，它还返回一个指针数组(pt)，等价于相等值线索。单元线索第 i 相的指针是 pt[i]，这里 i 是相控制区索引号 phase_domain_index。pt[i] 可以用作宏的自变

量，是第 i 相等值线索的指针。相控制区索引号 phase_domain_index 可以用宏 PHASE_DOMAIN_INDEX 恢复。

```
Thread **pt;
Thread *cell_threads;
Domain *mixture_domain;
mp_thread_loop_c(cell_threads, mixture_domain, pt)
```

mp_thread_loop_c 的自变量是 cell_threads、mixture_domain 和 pt。cell_threads 是网格线索的指针，mixture_domain 是 mixture-level 控制区的指针，pt 是含有 phase-level 线索的指针数组。

当要用包含控制区变量，如使用 DEFINE_ADJUST 的宏定义时，mixture_domain 通过 Fluent 的求解器自动传递给 UDF 文件，UDF 就和混合物相关了。若 mixture_domain 没有显式地传递给 UDF 文件，应用另外一个工具，如与前面相似的 Get_Domain(1)来恢复。注意：pt 和 cell_threads 的值是由查询函数派生出来的，mp_thread_loop_c 一般典型的用于 begin_c_loop 中，begin_c_loop 查询网格线索内的所有网格。当 begin_c_loop 嵌套在 mp_thread_loop_c 中，就可以查询混合物中相单元线索的所有网格了。

4. 查询混合物中所有的相面线索

mp_thread_loop_f 宏查询混合物控制区内所有混合物等值线索的面线索，并且给出每个与混合物等值线索有关的相等值线索指针。在混合物控制区内这和 thread_loop_f 宏几乎是等价的，其区别是：除了查找每一个面线索，这个宏还返回一个指针数组 pt，它与相等值线索相互关联。指向第 i 相的面线索指针是 pt[i]，这里是 phase_domain_index。当需要相等值线索指针时，pt[i] 可以作为宏的自变量。phase_domain_index 可以用 PHASE_DOMAIN_INDEX 宏恢复。

```
Thread **pt;
Thread *face_threads;
Domain *mixture_domain;
mp_thread_loop_f(face_threads, mixture_domain, pt)
```

mp_thread_loop_f 的自变量是 face_threads、mixture_domain 和 pt。face_threads 是面线索的指针，是混合物等值线索控制区的指针。pt 是包含相等值线索的指针数组。

当要用包含控制区变量，如 DEFINE_ADJUST 的宏来定义时，mixture_domain 通过 Fluent 的求解器自动传递给 UDF 文件，UDF 就和混合物相关了。若 mixture_domain 没有显式地传递给 UDF 文件，应用另外一个工具，如与前面类似的 Get_Domain(1)宏来恢复。注意：pt 和 cell_threads 的值是由查询函数派生出来的，mp_thread_loop_f 一般典型的用于 begin_f_loop 中，begin_f_loop 查询网格线索内的所有网格。当 begin_f_loop 嵌套在 mp_thread_loop_f 中，就可以查询混合物中相单元线索的所有网格了。

8.3.3 设置面变量

当要设置面的变量的值时，应用 F_PROFILE 宏。当要生成边界条件的外形或存储新的变量值时，自动调用这一函数。F_PROFILE 全部宏定义参考 mem.h 文件。

```
F_PROFILE( f, t, n)
```

F_PROFILE 的自变量是 f，面的索引号是 face_t； t，面上线索的指针，还有一个整数 n。 这些变量通过 Fluent 的求解器自动传递给用户编制的 UDF，不必给它们赋值。 整数 n 是要在边界上设定的变量标识符。例如，进口边界包含总压和总温，二者都在用户定义函数中定义。进口边界的变量在 Fluent 赋予整数 0，其他赋予 1。当在 Fluent 的进口边界对话框中定义边界条件时，这些整数值由求解器设定。

8.3.4　访问没有赋值的自变量

针对单相和多相的模型(如定义源项、性质和外形)，大多数标准的 UDF，函数所需要的变量(如 domain 或 thread pointers)在求解过程中通过求解器自动作为自变量直接传递给 UDF。例如，如果 UDF 定义了特殊边界区域(使用 DEFINE_PROFILE)的外形，在 Fluent 的边界条件对话框中，UDF 就和相混合物相关联了，在执行时通过求解器，合适的相混合物变量传递给函数。然而，并非所有的 UDF 都直接把函数所需要的自变量传递给求解器。回想一下，例如，DEFINE_ADJUST 和 DEFINE_INIT 传递给混合物控制区变量，这里 DEFINE_ON_DEMAND 是没有被传递的自变量。这部分提供了通过 DEFINE 函数访问没有被直接传递给 UDF 文件的工具。

1. Get_Domain

若控制区指针没有显式地作为自变量传递给 UDF，可以用 Get_Domain 宏恢复控制区指针。

```
Get_Domain(domain_id);
```

domain_id 是一个整数，混合物控制区其值为 1，在多相混合物模型中其值依次加 1。

注意： Get_Domain(1)用来代替以前的 Fluent 版本中外部变量 Domain *domain 表达式。

在单相流中， domain_id 的值为 1，Get_Domain(1)将返回流体控制区指针。

```
DEFINE_ON_DEMAND(my_udf)
{
 Domain *domain;          /* domain is declared as a variable  */
 domain = Get_Domain(1);  /* returns fluid domain pointer      */
 ...
}
```

在多相流中，Get_Domain 的返回值或者是混合物等值线索，或单相等值线索，或相等值线索，或相等值线索控制区指针。domain_id 的值在混合物控制区始终是 1，可以用 Fluent 里的图形用户界面获得 domain_id。简单地说，在 Fluent 的相对话框中，选择所需的相，domain_id 将被显示出来。需要用硬件代码整数 ID 作为自变量传递给宏。

```
DEFINE_ON_DEMAND(my_udf)
{
 Domain *mixture_domain;
 mixture_domain = Get_Domain(1);   /* returns mixture domain pointer */
                                   /* and assigns to variable       */
```

```
  Domain *subdomain;
  subdomain = Get_Domain(2); /* returns phase with ID=2 domain pointer*/
                             /* and assigns to variable             */
  …
}
```

下面是一个名为 get_coords 的 UDF 打印出了两个指定线索 ID 面的线面质心。这一函数执行在单相流中用的 Get_Domain 工具。在这个例子中，函数 Print_Thread_Face_Centroids 用了 Lookup_Thread 函数来决定线索的指针，然后输出了具体文件中线索的所有面的质心。调用 Get_Domain(1) 函数返回控制区的指针(或在多相流中是混合物控制区)。这一变量没有被传递给 DEFINE_ON_DEMAND。

```
#include "udf.h"
FILE *fout;
Print_Thread_Face_Centroids(Domain *domain, int id)
{
  real FC[2];
  face_t f;
  Thread *t = Lookup_Thread(domain, id);
  fprintf(fout,"thread id %d\n", id);
  begin_f_loop(f,t)
    {
     F_CENTROID(FC,f,t);
     fprintf(fout, "f%d %g %g %g\n", f, FC[0], FC[1], FC[2]);
    }
  end_f_loop(f,t)
  fprintf(fout, "\n");
}
DEFINE_ON_DEMAND(get_coords)
{
  Domain *domain;
  domain = Get_Domain(1);
  fout = fopen("faces.out", "w");
  Print_Thread_Face_Centroids(domain, 2);
  Print_Thread_Face_Centroids(domain, 4);
  fclose(fout);
}
```

2. 通过相控制区索引号使用相控制区指针

有两个方法可以获得混合物控制区具体相(或子区)的指针，即可以用宏 DOMAIN_SUB_DOMAIN (在下面描述)或 Get_Domain。DOMAIN_SUB_DOMAIN 有两个自变量：mixture_domain 和 phase_domain_index。这个函数返回给定 phase_domain_index 的相指针。注意：DOMAIN_SUB_DOMAIN 在执行上和 THREAD_SUB_THREAD 宏相似。

```
int phase_domain_index = 0; /* primary phase index is 0 */
Domain *mixture_domain;
Domain *subdomain = DOMAIN_SUB_DOMAIN(mixture_domain,phase_domain_index);
```

mixture_domain 是 mixture-level domain 的指针。

当用包含控制区自变量(如 DEFINE_ADJUST)的宏 DEFINE 时，自动通过 Fluent 的求解器传递给用户编制的 UDF 文件，UDF 文件就和混合物相关联了。否则，如果 mixture_domain 没有显式地传递给 UDF 文件，需要在调用 sub_domain_loop 之前，用另外一个宏工具来恢复(如 Get_Domain(1))。

phase_domain_index 是子区指针的索引号。它是一个整数，初始相值为 0，以后每相依次加 1。当要用包含相控制区变量(DEFINE_EXCHANGE_PROPERTY、DEFINE_VECTOR_EXCHANGE_PROPERTY)的 DEFINE 宏时，phase_domain_index 自动通过 Fluent 的求解器传递给用户编制的 UDF 文件，UDF 文件就和互相作用的相有联系了。否则，需要硬代码调用 DOMAIN_SUB_DOMAIN 宏给 phase_domain_index 指针赋值。如果多相流模型有两相，那么 phase_domain_index 初始相的值是 0，第二相的值为 1。然而，如果多相流模型中有更多的相，需要用 PHASE_DOMAIN_INDEX 宏来恢复与给定区域相关的 phase_domain_index。

3. 通过相控制区索引号使用相等值线索指针

THREAD_SUB_THREAD 宏可以用来恢复给定相控制区索引号的 phase-level thread (subthread) 指针。THREAD_SUB_THREAD 有两个自变量：mixture_thread 和 phase_domain_index。这一函数返回给定 phase_domain_index 的 phase-level 线索指针。注意：THREAD_SUB_THREAD 在执行上与 DOMAIN_SUB_DOMAIN 宏相似。

```
int phase_domain_index = 0; /* primary phase index is 0 */
Thread *mixture_thread; /* mixture-level thread pointer */
Thread *subthread = THREAD_SUB_THREAD(mixture_thread,phase_domain_index);
```

mixture_thread 是 mixture-level 线索的指针。当要用包含控制区自变量(如 DEFINE_PROFILE)的宏 DEFINE 时，自动通过 Fluent 的求解器传递给用户编制的 UDF 文件，UDF 文件就和混合物相关联了。否则，如果混合物控制线索指针没有显式地传递给 UDF，需要在调用 Lookup_Thread 宏之前，用另外一个宏工具来恢复(如 Get_Domain(1))。

phase_domain_index 是子区指针的索引号。它是一个整数，初始相值为 0，以后每相依次加 1。当要用包含相控制区索引号变量(DEFINE_EXCHANGE_PROPERTY、DEFINE_VECTOR_EXCHANGE_PROPERTY)的 DEFINE 宏时，phase_domain_index 通过 Fluent 的求解器自动传递给 UDF 文件，UDF 文件就和具体的相互作用的相互相关联了。否则，需要用硬代码改变宏 THREAD_SUB_THREAD 的 phase_domain_index 值。如果多相流模型中只有两相，那么 phase_domain_index 初始相的值是 0，第二个相的值为 1。然而，如果有更多的相，需要用 PHASE_DOMAIN_INDEX 宏来恢复与给定区域相关的 phase_domain_index。

4. 通过混合物等值线索使用相线索指针数组

THREAD_SUB_THREADS 宏可以用以恢复指针数组 pt，其元素包含相等值线索(子线索)的指针。THREADS_SUB_THREADS 有一个变量 mixture_thread。

```
Thread *mixture_thread;
```

```
Thread **pt;   /* initialize pt */
pt = THREAD_SUB_THREADS(mixture_thread);
```

mixture_thread 是 mixture-level thread 代表网格线索或面线索的指针。当要用包含线索变量(如 DEFINE_PROFILE)的 DEFINE 宏时，通过 Fluent 的求解器自动传递给 UDF 文件，这个函数就和混合物有关了。否则，如果混合物线索的指针没有显式地传递给 UDF 文件，那么需要用另一个方法来恢复。例如，可以用 Lookup_Thread 宏。

pt[i] 数组的元素是与第 i 相的相等值线索有关的值，这里 i 是 phase_domain_index。当想恢复网格具体相的信息时，可以用 pt[i]作为一些网格变量宏的自变量。例如，C_R(c,pt[i])可以用来返回网格点 c 第 i 相的密度。指针 pt[i]可以用 THREAD_SUB_THREAD 来恢复，用 i 作为自变量。phase_domain_index 可以用宏 PHASE_DOMAIN_INDEX 来恢复。

5. 通过相控制区指针调用混合物控制区指针

当 UDF 有权访问特殊的相等值线索(子区)指针时，可以用宏 DOMAIN_SUPER_DOMAIN 恢复混合物等值线索控制区指针。DOMAIN_SUPER_DOMAIN 含有一个变量 subdomain。注意：DOMAIN_SUPER_DOMAIN 在执行上和 THREAD_SUPER_THREAD 宏是非常相似的。

```
Domain *subdomain;
Domain *mixture_domain = DOMAIN_SUPER_DOMAIN(subdomain);
```

subdomain 是多相流混合物控制区相等值线索的指针。当要用包含控制区变量(如 DEFINE_ADJUST)的 DEFINE 宏时，通过 Fluent 的求解器，它可以自动传递给 UDF 文件，这个函数就会和混合物中的第一相和第二相相关联了。注意：在当前的 Fluent 版本中，DOMAIN_SUPER_DOMAIN 将返回与 Get_Domain(1)相同的指针。这样，如果 UDF 可以使用子区的指针，建议使用宏 DOMAIN_SUPER_DOMAIN 来代替 Get_Domain 宏，以避免将来的 Fluent 版本造成的不兼容问题。

6. 通过相线索指针使用混合物线索指针

当 UDF 有权访问某一条相线索的子线索指针，且想恢复混合物的等值线索指针时，可以使用宏 THREAD_SUPER_THREAD。该宏有一个自变量 subthread。

```
Thread *subthread;
Thread *mixture_thread = THREAD_SUPER_THREAD(subthread);
```

subthread 在多相流混合物中是一个特殊的相等值线索指针。当要使用包含线索变量(如 DEFINE_PROFILE)的 DEFINE 宏时，通过 Fluent 的求解器，它可以自动传递给 UDF 文件，这个函数就和混合物中的两相相互关联了。注意：在执行上包含线索变量的 DEFINE 宏和 THREAD_SUPER_THREAD 宏是非常相似的。

7. 通过区的 ID 使用线索指针

当想要在 Fluent 的边界条件对话框中恢复与给定区域 ID 相关的线索指针时，可以使用宏 Lookup_Thread 。例如，假设 UDF 需要对该区域的特殊线索操作(并非查找所有的线索)，可以用 DEFINE 宏定义 UDF 文件，而不用将线索指针(如 DEFINE_ADJUST)从 Fluent 的求

解器传递到 UDF 文件。UDF 还可以使用 Lookup_Thread 来获得想要的指针。这一过程分两步：首先，应该从 Fluent 的边界条件对话框中获得该区域的 ID；然后，需要使用硬代码作为自变量调用宏 Lookup_Thread。Lookup_Thread 返回与给定区域 ID 相关的线索指针，可以将线索指针赋给 thread_name，在 UDF 中使用。

```
int zone_ID;
Thread *thread_name = Lookup_Thread(domain,zone_ID);
```

在多相流的上下文中，通过宏 Lookup_Thread 返回的线索是与控制区自变量相关的相的等值线索。

下面是一个使用宏 Lookup_Thread 的 UDF 文件，在这个例子中，通过宏 Lookup_Thread 返回给定区域 ID 的线索指针，将它赋给线索。在 begin_f_loop 和 F_CENTROID 中使用这个线索指针查找与这个线索相关的所有的面，获得面的质心输出到文件。

```
#include "udf.h"
/* domain passed to Adjust function is mixture domain for multiphase*/
DEFINE_ADJUST(print_f_centroids, domain)
{
  real FC[2];
  face_t f;
  int ID = 1;
      /* Zone ID for wall-1 zone from Boundary Conditions panel */
  Thread *thread = Lookup_Thread(domain, ID);
  begin_f_loop(f, thread)
    {
     F_CENTROID(FC, f, thread);
     printf("x-coord = %f  y-coord = %f", FC[0], FC[1]);
    }
  end_f_loop(f, thread)
}
```

8. 使用相的控制区指针

当有权访问与给定相等值线索控制区指针的 domain_id 时，可以使用 DOMAIN_ID。DOMAIN_ID 有一个自变量 subdomain，它是相等值线索控制区的指针。控制区(混合物)的最大等值线索的 domain_id 的默认值是 1。即：如果被传递给 DOMAIN_ID 的控制区指针是混合物控制区的等值线索指针，那么，函数的返回值为 1。注意：当在 Fluent 的相对话框中选择需要的相时，宏所返回的 domain_id 是和显示在图形用户界面中的整数值 ID 相同的。

```
Domain *subdomain;
int domain_id = DOMAIN_ID(subdomain);
```

9. 通过相控制区使用相控制区索引号

宏 PHASE_DOMAIN_INDEX 返回给定相等值线索控制区(子区)指针的 phase_domain_index。PHASE_DOMAIN_INDEX 有一个自变量 subdomain，它是 phase-level domain

的指针。phase_domain_index 是子区指针的索引号，初始相的值为整数 0，以后每相依次加 1。

```
Domain *subdomain;
int phase_domain_index = PHASE_DOMAIN_INDEX(subdomain);
```

8.3.5 访问邻近网格和线索的变量

可以用 Fluent Inc. 提供的宏来确定邻近网格面。在复杂的 UDF 文件中，当查询特定网格或线索的面时，可能会用到这个信息。对给定的面 f 和它的线索 tf，两个相邻的网格点为 c0 和 c1。若是控制区附面层上的面则只有 c0，c1 的值为 NULL。一般地，当把网格导入到 Fluent 中时，按照右手定则定义面上节点的顺序，面 f 上的网格点 c0、c1 都存在。下面的宏返回网格点 c0 和 c1 的 ID 及所在的线索。

```
cell_t c0 = F_C0(f,tf);   /* returns ID for c0*/
   tc0 = THREAD_T0(tf);       /* returns the cell thread for c0 */
   cell_t c1 = F_C1(f,tf);    /*returns ID for c1*/
   tc1 = THREAD_T1(tf);       /* returns the cell thread for c1 */
```

注意：由 F_AREA 和 F_FLUX 返回的信息是直接相关的，这些值从网格 c0 到 c1 返回正值。

8.3.6 矢量工具

Fluent 提供了一些工具，可以用来在 UDF 中计算有关矢量的量。这些工具在源程序中以宏的形式运行。例如，可以用实函数 NV_MAG(V)计算矢量 V 的大小(模)，另外可以用函数 NV_MAG2(V)获得矢量 V 模的平方。下面是在 UDF 中可以利用的矢量工具列表。在矢量工具宏中有个约定俗成的惯例，V 代表矢量，S 代表标量，D 代表一系列三维的矢量，最后一项在二维计算中被忽略。

在矢量函数中约定的计算顺序括号、指数、乘除、加减(PEMDAS)不再适用。相反，下划线索符号(_)用来表示一组操作数，因此对元素的操作先于形成一个矢量。

1. NV_MAG

NV_MAG 计算矢量的大小，即矢量平方和的平方根。

```
NV_MAG(x)
   2D:  sqrt(x[0]*x[0] + x[1]*x[1]);
   3D:  sqrt(x[0]*x[0] + x[1]*x[1] + x[2]*x[2]);
```

2. NV_MAG2

NV_MAG2 计算矢量的平方和。

```
NV_MAG2(x)
   2D:  (x[0]*x[0] + x[1]*x[1]);
   3D:  (x[0]*x[0] + x[1]*x[1] + x[2]*x[2]);
```

3. ND_ND

在 RP_2D (Fluent 2D) 和 RP_3D (Fluent 3D)中，常数 ND_ND 定义为 2。如果想在 2D 中建立一个 2×2 矩阵，或在 3D 中建立一个 3×3 矩阵，可以用到这个量。当用 ND_ND 时，UDF 可以在 2D 和 3D 情况下使用，不用做任何改动。

```
real A[ND_ND][ND_ND]
for (i=0; i<ND_ND; ++i)
  for (j=0; J<ND_ND; ++j)
    A[i][j] = f(i, j);
```

4. ND_SUM

ND_SUM 计算 ND_ND 的和。

```
ND_SUM(x, y, z)
    2D:  x + y;
    3D:  x + y + z;
```

5. ND_SET

ND_SET 产生 ND_ND 任务说明。

```
ND_SET(u, v, w, C_U(c, t), C_V(c, t), C_W(c, t))
    u = C_U(c, t);
    v = C_V(c, t);
if 3D:
    w = C_W(c, t);
```

6. NV_V

NV_V 完成对两个矢量的操作。

```
NV_V(a, =, x);
    a[0] = x[0]; a[1] = x[1]; etc.
```

注意：如果在上面的方程中用 += 代替=，将得到 "a[0]+=x[0];" 等。

7. NV_VV

NV_VV 完成对矢量的基本操作。在下面的宏调用中，这些操作用符号-、/、*代替+。

```
NV_VV(a, =, x, +, y)
    2D:  a[0] = x[0] + y[0], a[1] = x[1] + y[1];
```

8. NV_V_VS

NV_V_VS 用来把两个矢量相加(后一项都乘一常数)。

```
NV_V_VS(a, =, x, +, y, *, 0.5);
    2D:  a[0] = x[0] + (y[0]*0.5), a[1] = x[1] +(y[1]*0.5);
```

注意：符号+ 可以换成-、*或/，符号*可以替换成/。

9. NV_VS_VS

NV_VS_VS 用来把两个矢量相加(每一项都乘一常数)。

```
NV_VS_VS(a, =, x, *, 2.0, +, y, *, 0.5);
    2D:  a[0] = (x[0]*2.0) + (y[0]*0.5), a[1] = (x[1]*2.0) + (y[1]*0.5);
```

注意： 符号+可以换成-、*、或/，符号*可以替换成/。

10. ND_DOT

下列的工具计算两个矢量的点积。

```
ND_DOT(x, y, z, u, v, w)
    2D:  (x*u + y*v);
    3D:  (x*u + y*v + z*w);
NV_DOT(x, u)
    2D:  (x[0]*u[0] + x[1]*u[1]);
    3D:  (x[0]*u[0] + x[1]*u[1] + x[2]*u[2]);
NVD_DOT(x, u, v, w)
    2D:  (x[0]*u + x[1]*v);
    3D:  (x[0]*u + x[1]*v + x[2]*w);
```

8.4 UDF 常用 DEFINE 宏

DEFINE 宏一般包括通用求解宏、模型指定宏、多相流模型宏、离散相模型宏。这里主要介绍与本书相关的通用求解宏、模型指定宏、多相流模型宏。

8.4.1 通用求解宏

(1) DEFINE_ADJUST 是一个调节和修改 Fluent 变量的通用宏。例如，可以用 DEFINE_ADJUST 来修改流动变量(如速度、压力)并计算积分，也可以用它来对某一标量在整个流场上积分，然后在该结果的基础上调节边界条件。在每一步迭代中都可以执行用 DEFINE_ADJUST 定义的宏，并在解输运方程之前的每一步迭代中调用它。其宏格式如下：

```
DEFINE_ADJUST(name, d);
```

DEFINE_ADJUST 有两个参变量：name 和 d。name 是所指定的 UDF 的名字，当 UDF 编译并连接时， Fluent 图形用户界面就会显示这个名字，此时就可以选择它了；d 是 Fluent 解算器传给 UDF 的变量，它是一个指向区域的指针，调节函数被应用于这个区域上。区域变量提供了存取网格中所有单元和表面的线程。对于多相流，由解算器传给函数的区域指针是混合层区域指针。DEFINE_ADJUST 函数不返回任何值给解算器。

(2) DEFINE_INIT 是用来定义一组解的初始值。每一次初始化时 DEFINE_INIT 函数都会被执行一次，并在解算器完成默认的初始化之后立即被调用。因为它是在流场初始化之后被调用的，所以它常用于设定流动变量的初值。其宏格式如下：

```
DEFINE_INIT(name, d)
```

DEFINE_INIT 有两个参变量：name 和 d。name 是所指定的 UDF 的名字，当 UDF 编译并连接时，Fluent 图形用户界面就会显示这个名字，此时就可以选择它了；指针 d 是 Fluent 解算器传给 UDF 的变量。

(3) DEFINE_ON_DEMAND 宏用来定义在 Fluent 中执行命令的 UDF。其格式如下：

```
DEFINE_ON_DEMAND ( name );
```

该宏只有一个参数，即 UDF 名字，当编译和连接后，函数名会在 Fluent 图形用户界面中变得可见，能被选择。

(4) DEFINE_RW_FILE 宏用来定义用户想写入 cas 或 dat 文件的信息。其格式如下：

```
DEFINE_RW_FILE ( name, fp);
```

这个宏的参数有 name 和 fp 两个。name 是函数名，与前面一样，当编译和连接后，函数名会出现在 Fluent 图形用户界面中，且能被选择；fp 是文件指针，指向所需写的文件。

8.4.2 模型指定宏

在 Fluent 中，模型指定宏有 14 个，这里仅介绍比较常用，特别是与动网格相关的宏。

(1) DEFINE_PROFILE 宏定义用户边界变量在边界上随边界坐标或随时间的变化，如速度、压力、温度、体积分数、组分分数等随实际或坐标变化的边界条件。其格式如下：

```
DEFINE_PROFILE ( name, t, i);
```

这个宏有三个参数：name、t、i。name 是函数名，与前面一样，当编译和连接后，函数名会出现在 Fluent 图形用户界面中，且能被选择；指针 t 指向所需施加边界条件的边界；i 是用来识别已被定义变量的索引。

(2) DEFINE_PROPERTY 宏用来定义材料属性，如密度、粘性系数、热传导、质量扩散等。其格式如下：

```
DEFINE_PROPERTY ( name, c, t);
```

它的三个参数是 name、c 和 t。name 是函数名，当编译和连接后，函数名会出现在 Fluent 图形用户界面中，且能被选择；c 是识别单元是否在所给线索内的索引；t 是指向已定义材料属性的线索指针。

(3) DEFINE_SOURCE 用来定义 Fluent 输运方程中的源项，包括连续性、动量方程、k、ε、能量、组分质量分数等。其格式如下：

```
DEFINE_SOURCE ( name, c, t, dS, eqn);
```

这个宏中，除了 name 外，还有较多的参数，如 c、t、dS、eqn。c 是 t 所指向的线索上的单元索引；数组 dS 说明了相对于输运方程的应变量的源项导数；eqn 为数组元素的编号。对于 source$=-0.5C_2\rho_y|v_x||v_y|$ 的 x 方向动量源项 UDF 如下：

```
#include "udf.h"
#define C2 100.0
DEFINE_SOURCE(xmom_source, c, t, dS, eqn)
```

```
{
real x[ND_ND];
real con, source;
C_CENTROID(x, c, t);
con = C2*0.5*C_R(c, t)*x[1];
source = -con*fabs(C_U(c, t))*C_U(c, t);
dS[eqn] = -2.*con*fabs(C_U(c, t));
return source;
}
```

(4) DEFINE_CG_MOTION 宏用来定义刚体重心运动。其定义格式如下：

```
DEFINE_CG_MOTION ( name, dt, vel, omega, time, dtime) ;
```

在该宏中，*dt 是动网格线索指针；vel[]是返回给 CG 速度的数组；omega[]是返回给 CG 的角速度；time 是时间；dtime 是时间步。例如，根据牛顿第二运动定律给定的运动速度 UDF 如下：

```
#include "udf.h"
#include "dynamesh_tools.h"
static real v_prev = 0.0;
DEFINE_CG_MOTION(piston, dt, vel, omega, time, dtime)
{
 Thread *t;
 face_t f;
 real NV_VEC (A);
 real force, dv;
 /* reset velocities */
 NV_S (vel, =, 0.0);
 NV_S (omega, =, 0.0);
 if (!Data_Valid_P ())
  return;
 /* get the thread pointer for which this motion is defined */
 t = DT_THREAD ((Dynamic_Thread *)dt);
/* compute pressure force on body by looping through all faces */
 force = 0.0;
 begin_f_loop (f, t)
  {
   F_AREA (A, f, t);
   force += F_P (f, t) * NV_MAG (A);
  }
 end_f_loop (f, t)
 /* compute change in velocity, i.e., dv = F * dt / mass velocity update
using explicit euler formula */
 dv = dtime * force / 50.0;
 v_prev += dv;
 Message ("time = %f, x_vel = %f, force = %f\n", time, v_prev,
 force);
 /* set x-component of velocity */
 vel[0] = v_prev;
```

```
}
```

(5) DEFINE_GEOM 宏用来定义变形区域的几何形状。其格式如下：

```
DEFINE_GEOM ( name, d, dt, position);
```

在该宏中，*d 为区域指针；*dt 为动网格线索指针；*position 为边界节点位置指针。例如，定义 parabola 边界通过点(0, 1)、(1/2, 5/4)、(1, 1)的函数如下所示：

```
#include "udf.h"
DEFINE_GEOM(parabola, domain, dt, position)
{
  /* set y = -x^2 + x + 1 */
  position[1] = - position[0]*position[0] + position[0] + 1;
}
```

(6) DEFINE_GRID_MOTION 宏定义单个节点位置(如在流体结构相互作用的节点位置)。其格式如下：

```
DEFINE_GRID_MOTION ( name, d, dt, time, dtime);
```

该宏中，*d 为区域指针；*dt 为动网格线索指针；time 是时间；dtime 是时间步。对于下式确定的变形梁的变形函数如下所示。

$$\omega(x,t)=\begin{cases}-10.4\sqrt{x}\sin(27.178t), x>0.02\\ 0, \quad x<0.02\end{cases}$$

```
#include "udf.h"
DEFINE_GRID_MOTION(beam, domain, dt, time, dtime)
{
  Thread *tf = DT_THREAD (dt);
  face_t f;
  Node *v;
  real NV_VEC (omega), NV_VEC (axis), NV_VEC (dx);
  real NV_VEC (origin), NV_VEC (rvec);
  real sign;
  int n;
    /* set deforming flag on adjacent cell zone */
  SET_DEFORMING_THREAD_FLAG (THREAD_T0 (tf));
  sign = -5.0 * sin (26.178 * time);
  Message ("time = %f, omega = %f\n", time, sign);
 NV_S (omega, =, 0.0);
  NV_D (axis, =, 0.0, 1.0, 0.0);
  NV_D (origin, =, 0.0, 0.0, 0.152);
  begin_f_loop (f, tf)
    {
      f_node_loop (f, tf, n)
        {
          v = F_NODE (f, tf, n);
          /* update node if x position is greater than 0.02
             and that the current node has not been previously
             visited when looping through previous faces */
```

```
        if (NODE_X (v) > 0.020 && NODE_POS_NEED_UPDATE (v))
          {
           /* indicate that node position has been update
             so that it's not updated more than once */
           NODE_POS_UPDATED (v);
           omega[1] = sign * pow (NODE_X (v)/0.230, 0.5);
           NV_VV (rvec, =, NODE_COORD (v), -, origin);
           NV_CROSS (dx, omega, rvec);
           NV_S (dx, *=, dtime);
           NV_V (NODE_COORD (v), +=, dx);
          }
       }
    }
  end_f_loop (f, tf);
}
```

8.4.3 多相流模型宏

(1) DEFINE_CAVITATION_RATE 宏定义由于多相流压力张力产生的蒸汽模型。其格式如下：

```
DEFINE_CAVITATION_RATE ( name, c, t, p, rhoV, rhoL, vofV, p_v, n_b, m_dot)
```

该宏中，t 是混合水平线索指针；c 是 t 指向的线索上的单元索引；其余参数是实指针，分别指向下列数据：共享压力 p，蒸汽密度 rhoV，液体密度 rhoL，蒸汽体积分数 vofV，蒸汽压 p_v，每单位体积内的气泡数目 n_b，蒸汽的形成率 m_dot。举例如下：

```
#include "udf.h"
#define MIN_VOF 1.e-5
#define MAX_VOF 0.999999
DEFINE_CAVITATION_RATE(cavitation_rate, c, t, p, rhoV, rhoL, vofV, p_v, n_b,
m_dot)
{
   real p_vapor = *p_v;
   real n_bubbles = *n_b;
   real dp, vofM, radV;
   dp = p_vapor - ABS_P( p[c], op_pres );
   vofM = MIN(MAX(MIN_VOF, vofV[c]),MAX_VOF);
   radV = pow(vofM/((1.-vofM)*4./3.*M_PI*n_bubbles), 1./3.);
   if (dp>0.)
      *m_dot = (1.-vofV[c]) * n_bubbles * 4. * M_PI *
      radV * radV/(1.+n_bubbles*4./3.*M_PI*radV*radV*radV) *
      sqrt(2.*ABS(dp)/(3.*rhoL[c]));
   else
   {
      *m_dot = - (1.-vofV[c]) * n_bubbles * 4. *M_PI *
      radV*radV/(1.+n_bubbles*4./3.*M_PI*radV*radV*radV) *
      sqrt(2.*ABS(dp)/(3.*rhoL[c]));
      if (vofV[c] <= MIN_VOF) *m_dot=0.;
```

```
    }
}
```

(2) DEFINE_EXCHANGE_PROPERTY 宏用来定义欧拉模型多相流中的升力和拖曳系数。其格式如下：

```
DEFINE_EXCHANGE_PROPERTY ( name, c, mixture_thread,
second_column_phase_index, first_column_phase_index);
```

该宏中，mixture_thread 是混合水平线索的指针；c 是 mixture_thread 指针所指的线索上的单元索引；first_column_phase_index 和 second_column_phase_index 为有相互滑移速度的一对相的相应整数标识，这个标识在图形用户界面的 Phase Interaction(相相互作用)对话框中选择对应的相。对于名称为 custom_drag 的修改默认的 Syamlal 拖曳定律的函数，修改最小的流体速度为 8cm/s，bfac 参数为 0.28 和 9.07。具体函数描述如下：

```
#include "udf.h"
#define pi 4.*atan(1.)
#define diam2 3.e-4
DEFINE_EXCHANGE_PROPERTY(custom_drag, cell, mix_thread, s_col, f_col)
{
Thread *thread_g, *thread_s;
real x_vel_g, x_vel_s, y_vel_g, y_vel_s, abs_v, slip_x, slip_y, rho_g, rho_s,
mu_g, reyp, afac, bfac, void_g, vfac, fdrgs, taup, k_g_s;
/* find the threads for the gas (primary) */
/* and solids (secondary phases) */
thread_g = THREAD_SUB_THREAD(mix_thread, s_col);/* gas phase */
thread_s = THREAD_SUB_THREAD(mix_thread, f_col);/* solid phase*/
/* find phase velocities and properties*/
x_vel_g = C_U(cell, thread_g);
y_vel_g = C_V(cell, thread_g);
x_vel_s = C_U(cell, thread_s);
y_vel_s = C_V(cell, thread_s);
slip_x = x_vel_g - x_vel_s; slip_y = y_vel_g - y_vel_s; rho_g = C_R(cell,
thread_g);
rho_s = C_R(cell, thread_s);
mu_g = C_MU_L(cell, thread_g); /*compute slip*/
abs_v = sqrt(slip_x*slip_x + slip_y*slip_y); /*compute reynolds number*/
reyp = rho_g*abs_v*diam2/mu_g; /* compute particle relaxation time */
taup = rho_s*diam2*diam2/18./mu_g; void_g = C_VOF(cell, thread_g);/* gas vol
frac*/ /*compute drag and return drag coeff, k_g_s*/
afac = pow(void_g,4.14);
if(void_g<=0.85) bfac = 0.281632*pow(void_g, 1.28);
else bfac = pow(void_g, 9.076960);
vfac = 0.5*(afac-0.06*reyp+sqrt(0.0036*reyp*reyp+0.12*reyp*(2.*bfac-
afac)+afac*afac));
fdrgs = void_g*(pow((0.63*sqrt(reyp)/ vfac+4.8*sqrt(vfac)/vfac),2))/24.0;
k_g_s = (1.-void_g)*rho_s*fdrgs/taup; return k_g_s;
}
```

(3) DEFINE_VECTOR_EXCHANGE_PROPERTY 宏用来说明多相流混合模型的滑移速度。其格式如下：

```
DEFINE_VECTOR_EXCHANGE_PROPERTY ( name, c, mixture_thread,
second_column_phase_index, first_column_phase_index, vector_result)
```

该宏中，mixture_thread 是混合水平线索的指针；c 是 mixture_thread 指针所指的线索上的单元索引；first_column_phase_index 和 second_column_phase_index 为有相互滑移速度作用的一对相的相应整数标识，这个标识在图形用户界面的 Phase Interaction(相相互作用)对话框中选择对应的相。两相滑移速度 UDF 实例如下：

```
#include "udf.h"
DEFINE_VECTOR_EXCHANGE_PROPERTY(custom_slip, c, mixture_thread,
 second_column_phase_index, first_column_phase_index, vector_result)
{
real grav[2] = {0., -9.81};
real K = 5.e4; real pgrad_x, pgrad_y;
Thread *pt, *st;/* thread pointers for primary and secondary phases*/
pt = THREAD_SUB_THREAD(mixture_thread, second_column_phase_index);
st = THREAD_SUB_THREAD(mixture_thread, first_column_phase_index);
/* at this point the phase threads are known for primary (0) and secondary(1)
phases */
pgrad_x = C_DP(c, mixture_thread)[0];
pgrad_y = C_DP(c, mixture_thread)[1];
vector_result[0] = -(pgrad_x/K) +( ((C_R(c, st)- C_R(c, pt))/K)* grav[0]);
vector_result[1] = -(pgrad_y/K) +( ((C_R(c, st)- C_R(c, pt))/K)* grav[1]);
}
```

8.5 UDF 的解释和编译

要将编写好的 UDF 加入到 Fluent 中运行，需要先对 UDF 进行解释或者编译，然后再加载到 Fluent 中。

8.5.1 UDF 的解释运行

对 UDF 最简单的处理办法就是让 UDF 解释运行。其优点是不需要安装 C 编译器，适合于 UDF 程序较短的场合。

假如有如下 UDF，使某个进口面的速度为 10：

```
#include "udf.h"
DEFINE_PROFILE(inlet_velocity, thread, position)
{
  face_t f;
  begin_f_loop(f, thread)
  {
   F_PROFILE(f, thread, position) = 10;
  }
```

```
    end_f_loop(f, thread)
}
```

程序参考程序实例 ch09\9-1.c。

可以在文本编辑软件(如 Windows 自带的记事本)中打开并修改上面的程序。

运行 Fluent，选择 2d 版本，加载网格文件 ch09\9-1.cas，然后选择 Define→User-Defined→Functions→Interpreted 命令，弹出如图 8-1 所示的 Interpreted UDFs(UDF 解释)对话框。

在 Source File Name 文本框中输入 UDF 文件名，或者通过单击 Browse 按钮查找 UDF 文件，单击 Interpret 按钮，就开始了文件的解释过程。如程序有错，则 Fluent 会提示错误的原因。如程序正确，则解释后的程序自动加载到系统中，并将保存在工程的 cas 文件中。

选择 Define→Boundary→Conditions 命令，弹出如图 8-2 所示的 Boundary Conditions(边界条件)对话框；在 Zone 列表框中选择 in，在 Type 列表框中选择 velocity-inlet，单击 Set 按钮，弹出如图 8-3 所示的 Velocity Inlet(速度入口边界设置)对话框；从这个对话框中可以看到解释过后的 UDF。



图 8-1　Fluent 的 UDF 解释对话框

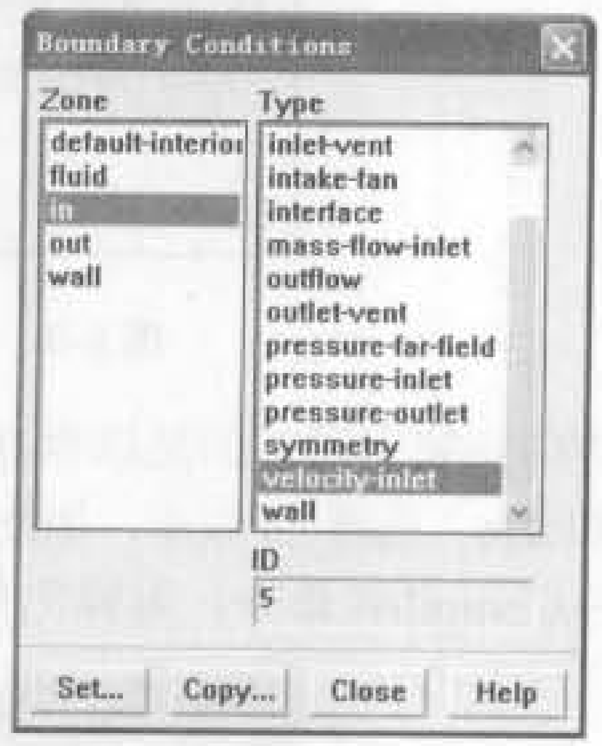


图 8-2　边界条件对话框

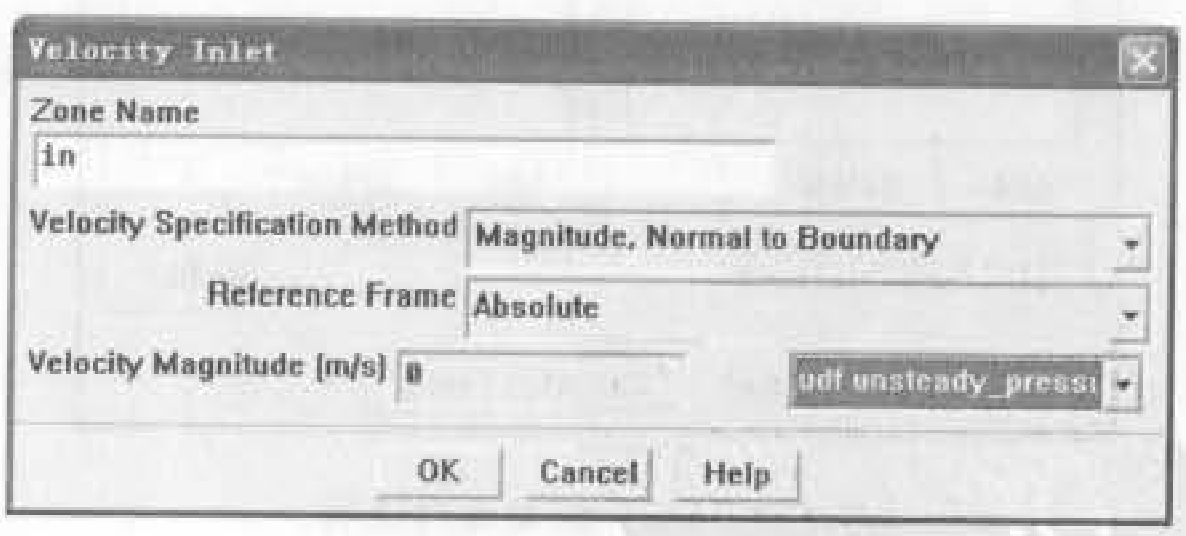


图 8-3　速度入口边界设置对话框

8.5.2　UDF 的 C 编译

UDF 可以解释运行，但是其运行效率较低，且很多的 UDF 函数不支持解释运行，故有必要掌握 UDF 的编译运行方法。

编译运行 UDF 的前提条件是正确安装了 C/C++编译器。推荐采用微软的 VC++编译器。在本书中，假定正确安装了 VC++6.0 编译器。

测试 VC++是否正确安装，且可以编译 UDF 的一个方法是在 DOS 命令窗口中运行 cl.exe 命令，如提示编译器的用法，表示安装正确，如图 8-4 所示；如提示 bad command，则表示 VC 安装有误。

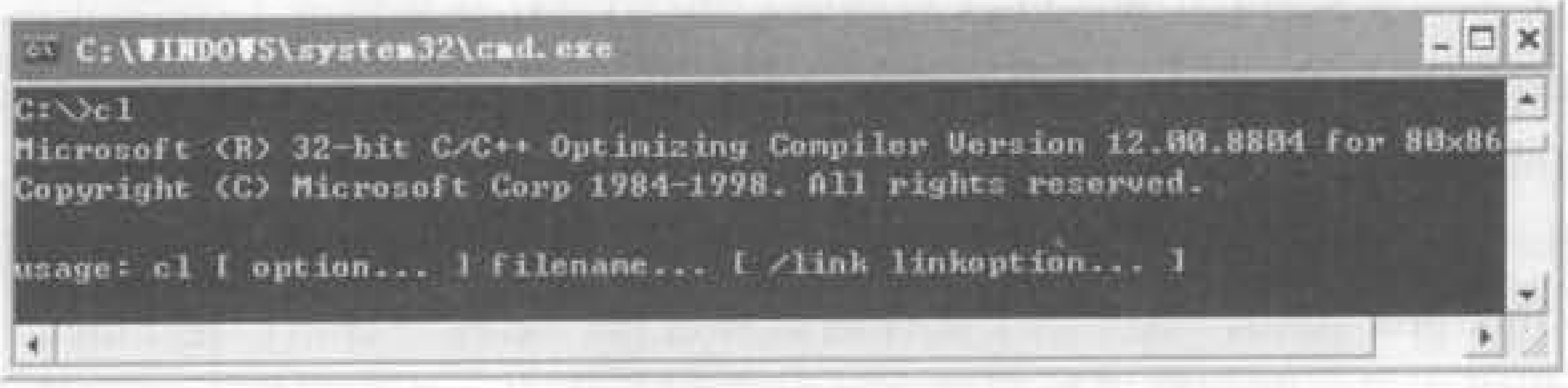


图 8-4　在 DOS 窗口中测试 VC 是否安装正确

如果在 DOS 下运行 cl.exe 出错，可以运行光盘中的“解决 VC 安装问题.exe”程序，修正 VC 安装中的错误，使在 DOS 下可以正确编译程序。该软件运行界面如图 8-5 所示。

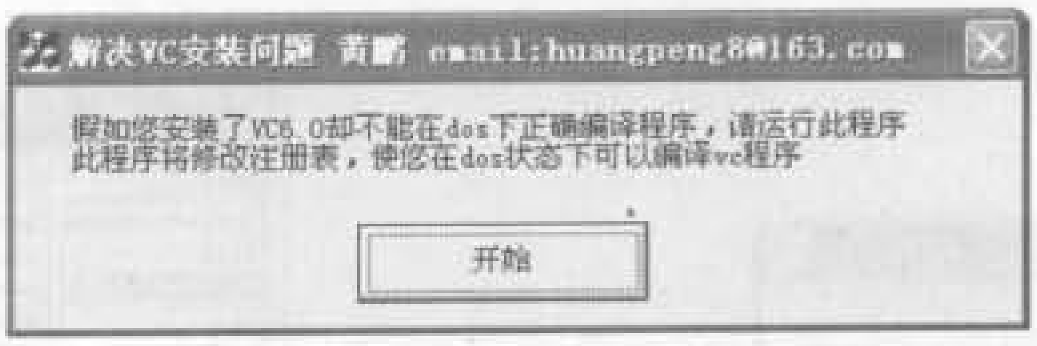


图 8-5　用软件修正 VC 安装中的错误

假如 VC 安装正确，仍然以程序 ch09\9-1.c 为例说明 UDF 的编译和加载过程。

运行 Fluent，选择 2d 版本，加载网格文件 ch09\9-1.cas，然后选择 Define→User-Defined→Functions→Compiled 命令，则弹出如图 8-6 所示的 Compiled UDFs(UDF 编译)对话框。

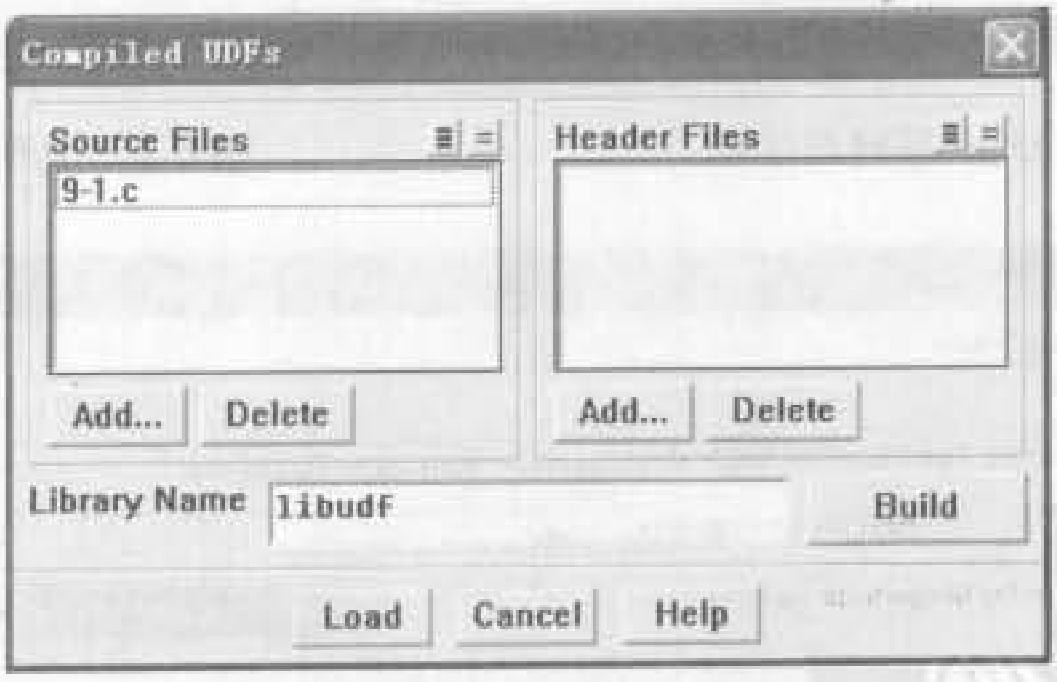


图 8-6　UDF 的编译对话框

左边的列表框用于增加和显示 UDF 程序，右边的列表框用于增加和显示需要的头文件。单击左边的 Add 按钮，加载 ch09\9-1.c 文件；单击 Build 按钮，则程序开始编译，编译信息显示在 Fluent 显示窗口中。如编译正确，单击 Load 按钮，则编译后的 UDF 加载到 Fluent 中，并随 cas 文件一起保存。

注意：UDF 程序 C 文件和 cas 文件一定要放在同一个目录中。

调出如图 8-3 所示的对话框，可以判断 UDF 是否已经正确加载。

8.5.3 UDF 的 VC++编译

用 C 语言编写小型的 UDF 比较方便，但是如 UDF 程序比较长，则最好采用 VC++编译程序。用 VC++编译 UDF 有如下优点。

- 采用 VC++的集成开发环境编程，程序的输入、编辑和编译过程比较容易。
- 用面向对象的 C++编程方法，编写程序比面向过程的 C 语言容易。当然也可以用 C 语言编写程序。
- 编程中可以使用类库 MFC、其他公司的类库如 AutoDesk 的 C++类库，或者用户自己的 C++类和类库。

Fluent 公司没有提供 UDF 的 C++编译方法。UDF 的 VC++编译程序 Fluent_VC_Programmer 在光盘中，也可以在流体中文网上下载。具体链接为

http://cfluid.imcas.net/cgi-bin/LB5000/topic.cgi?forum=51&topic=78

该软件目前只支持 VC++6.0。用户可以以它为工具，在 Windows 系统中编辑和编译 Fluent 并行或者非并行程序。该软件目前支持 Fluent 6.0、6122、6216、6313 版。软件运行的初始界面如图 8-7 所示。

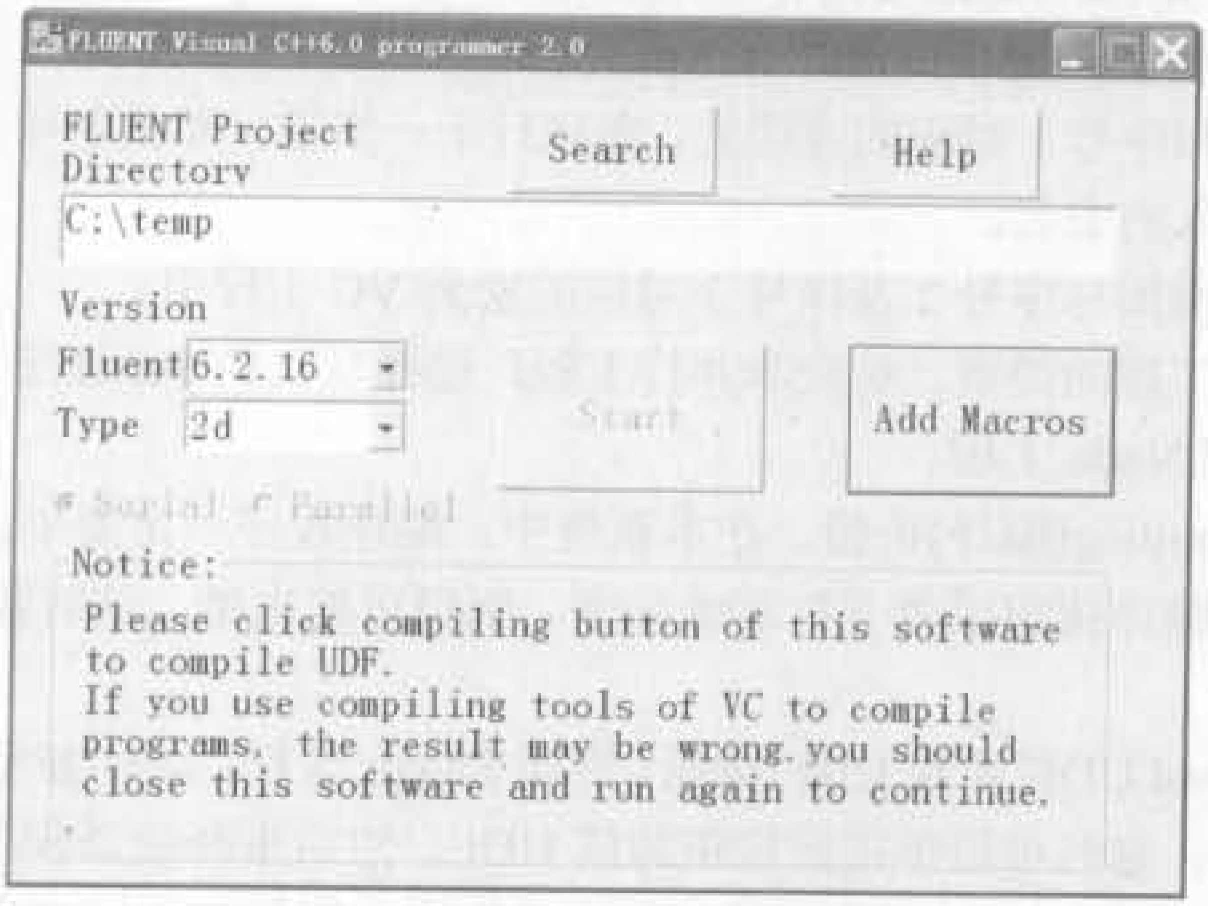


图 8-7　Fluent_VC_Programer 的初始界面

Fluent Project Directory 是工程目录，也就是 cas 或者.msh 文件所在的目录。后面所有的程序也将放到这个目录中。这个目录可以在文本框中填写，也可以通过 Search 按钮查找。C/C++文件将写在该目录。软件运行时，将搜索系统中所有的 Fluent 版本，版本号显示在下拉列表框中。这里假设如果系统中有多个 Fluent 版本，所有版本安装在同一个目录(默认为 C:\fluent.inc)下。Type 表示 Fluent 计算模型版本，有 2d、2ddp、3d、3ddp 几种。假如要求并行计算，选择并行计算，否则选择非并行。如 Start 按钮可用，表示项目目录里没有 VC++工程，单击 Start 按钮后将生成 VC++工程文件并用 VC++打开工程，假如 Start 按钮不可用，Add Macros 按钮可用，表示项目目录里有 VC++工程，单击 Add Macros 按钮将打开工程，用户可以修改和编译工程里的程序。

单击 Start 或者 Add Macros 按钮后，软件的界面变为如图 8-8 所示。

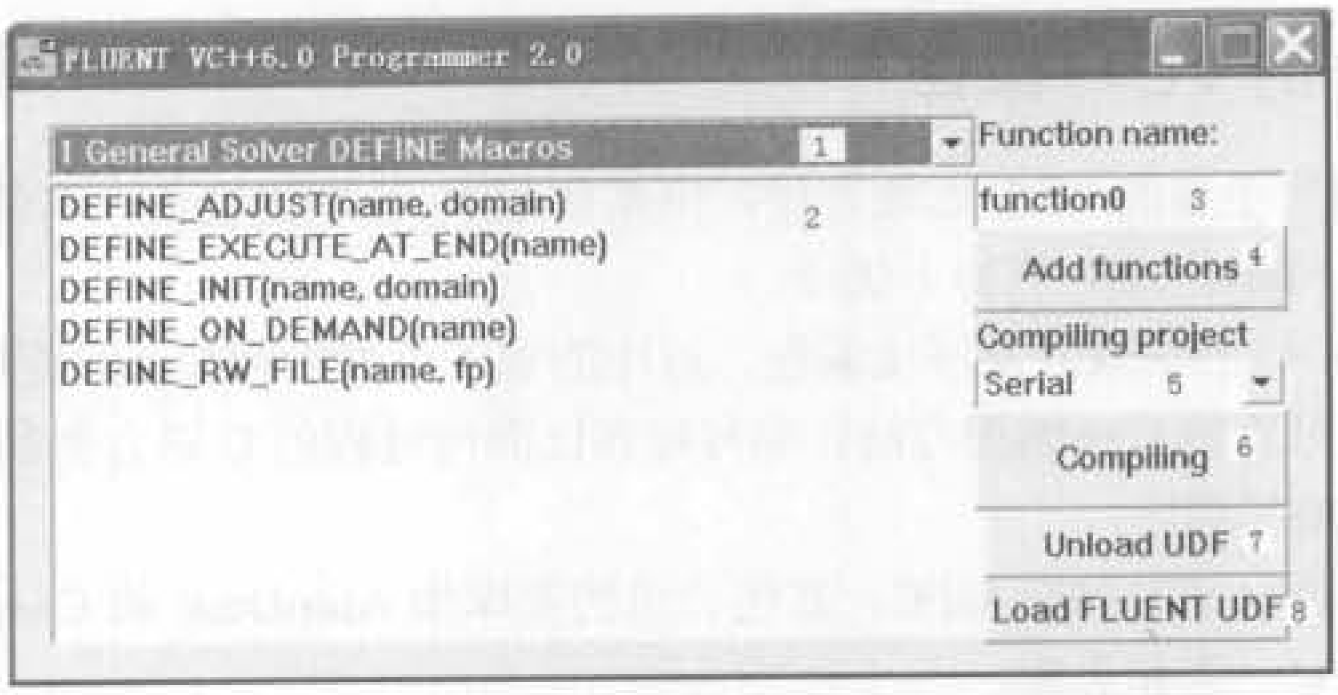


图 8-8 软件的界面

在图 8-8 中，序号 1 为下拉列表框，框中包含如下的几类宏。

- 1 General Solver DEFINE Macros
- 2 Model-Specic DEFINE Macros
- 3 Multiphase DEFINE Macros
- 4 Dynamic Mesh Model DEFINE Macros
- 5 Discrete Phase Model DEFINE Macros

序号 2 显示了和序号 1 对应的可用宏，在进行下一步前，要在序号 2 中选定需要的宏。

序号 3 为要输入的宏名。

序号 4 表示要增加与序号 2 及序号 3 对应的宏到 VC 工程中。

序号 5 为编译工程的类型。对应非并行工程，选择默认的 Serial 工程；对应并行工程，要分别编译 Host 和 Node 工程。

序号 6 为 Compiling(编译)按钮。在本系统中，编译程序一定要单击这个按钮。用 VC 本身的编译工具如编译按钮或者下拉菜单编译，则很可能出错。这时需要关闭本软件，然后重新打开运行。

序号 7 为 Unload UDF(卸载 UDF)按钮。假如 Fluent 运行，且 UDF 已经加载到了 Fluent 中，则不能再编译，这时可以单击此按钮卸载 UDF，然后重新编译程序。

序号 8 为 Load Fluent UDF(加载 Fluent UDF)按钮。假如 Fluent 运行，则可以单击此按钮加载 UDF。

下面是软件使用的实例。

(1) 打开 Fluent_VC_Programmer，如图 8-7 所示，在项目目录文本框中输入 e:\temp，选择合适的 Fluent 版本，选择非并行版本 2d 版本，单击 Start 按钮，这时，在 e:\temp 目录中创建了 VC++工程，并自动打开该工程；双击左边的 functions.cpp 文件，显示如图 8-8 所示。过了约 10 秒钟，Fluent_VC_Programmer 程序的主界面变成如图 8-9 所示。

(2) 在序号 1 中选择 2 Model-Specic DEFINE Macros，在序号 2 中选择 DEFINE_PROFILE(name,t,i)，函数名为 inlet_velocity，如图 8-10 所示。单击 Add functions 按钮，在出现的对话框中单击"是"按钮，则在 VC 工程中生成宏 DEFINE_PRANDTL_K(inlet_velocity, c, t)。将鼠标指针放到 VC 工程中，可能提示一个对话框，单击"是"按钮，则 VC 工程如图 8-11 所示。

在宏中增加代码后宏如下：

```
DEFINE_PROFILE(inlet_velocity, thread, position)
{
 face_t f;
 begin_f_loop(f, thread)
 {
  F_PROFILE(f, thread, position) = 10;
 }
 end_f_loop(f, thread)
}
```

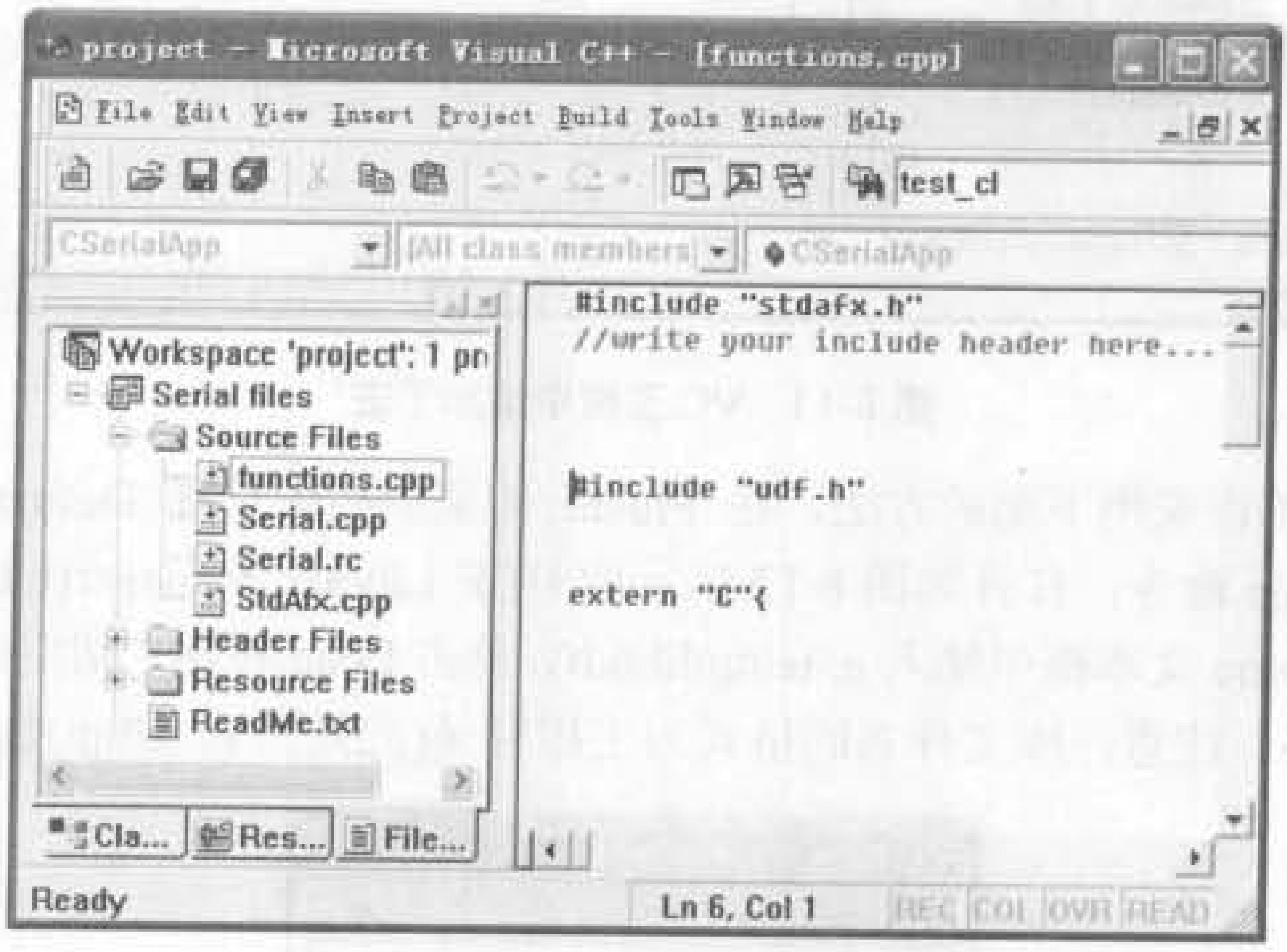


图 8-9 初始 VC++工程

(3) 这时软件中如图 8-10 中的 Compiling(编译)按钮已经可用，单击该按钮。开始编译程序，并使用了 VC 编译的结果。生成的程序的路径为 e:\temp\libudf\ntx86\2d\libudf.dll。

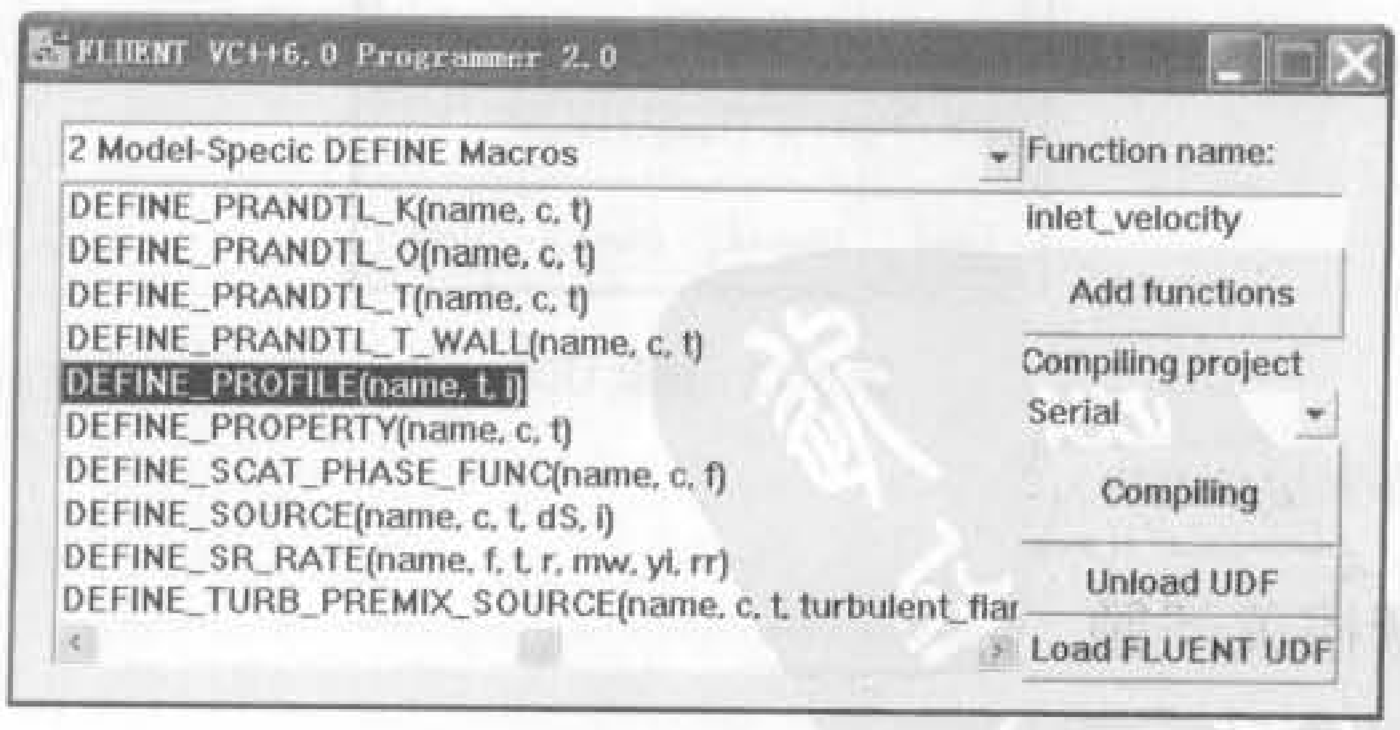


图 8-10 函数生成窗口

(4) 将光盘中的文件 ch09\9-5-1.cas 复制到 e:\temp 目录下。运行 Fluent，加载 cas 文件。

(5) 单击软件中的 Load(加载)Fluent UDF 按钮加载编译后的 UDF 程序。

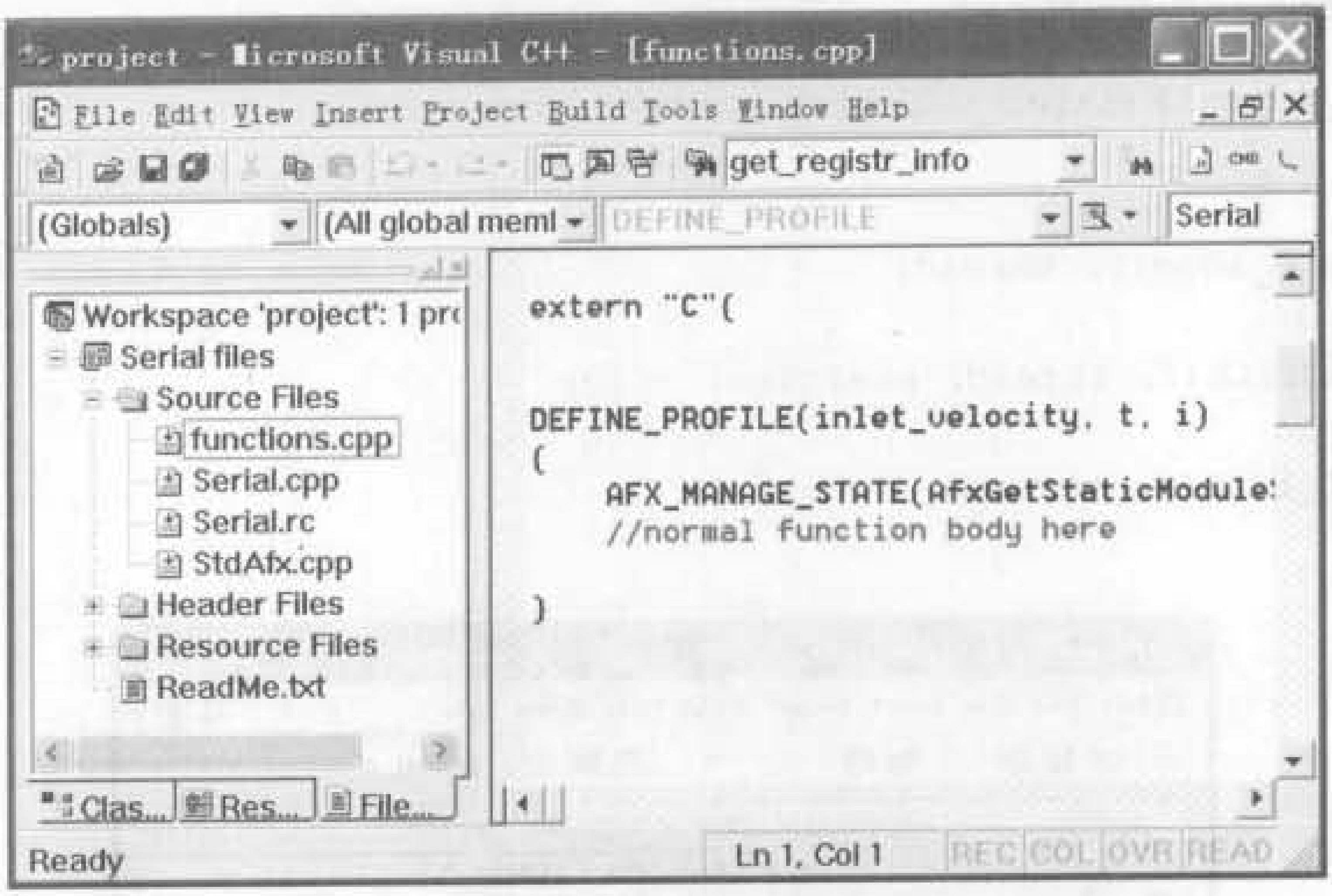


图 8-11 VC 工程中增加了宏

加载程序也可以采用下面的方法。在 Fluent 的菜单栏中选择 Define→User-Defined→Functions→Manage 命令，打开如图 8-12 所示的 UDF Library Manager(UDF 库管理器)对话框；在 Library Name 文本框中输入 e:\temp\libudf，单击 Load 按钮，则完成库文件加载，可以使用此 UDF 了。注意：库文件名的格式为工程目录(此处为 e:\temp\)加上 libudf。

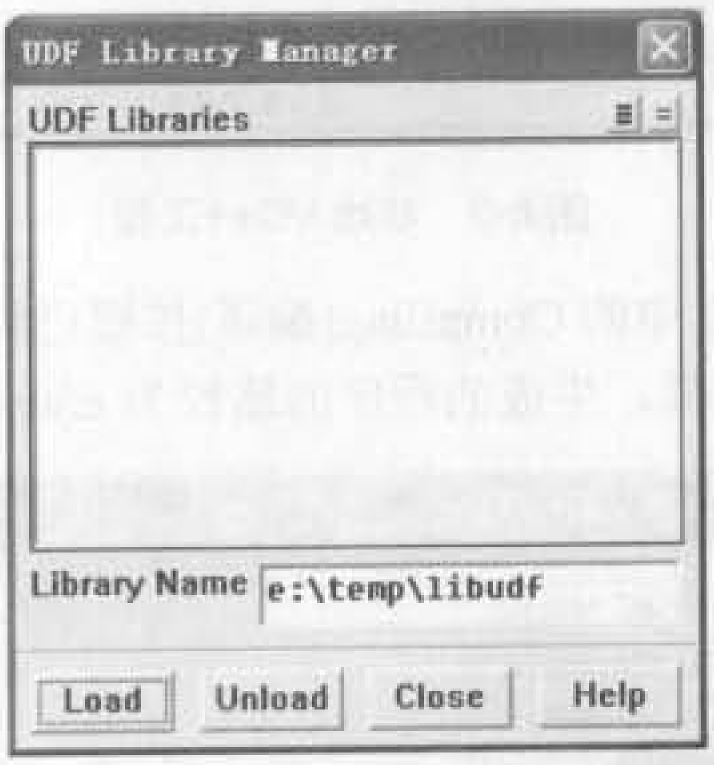


图 8-12 加载 UDF

至此，编译和加载成功。

8.5.4 编译相关问题

1. 关于头文件

一般来说，C++的 cpp 文件中要包含很多的头文件，就是那些以.h 做扩展名的文件。在本系统中，要求 C++、MFC 及其他函数的头文件在前，Fluent 的头函数在后，而不能相反。如：

```
#include "stdafx.h" //VC的标准头文件
#include "view.h"   // 自定义的头文件
//…                 //其他头文件
#include "udf.h"    //Fluent的头文件
```

2. 关于全局变量和宏

Fluent UDF已经定义了一些全局变量如x、y、z，假如程序中再定义和这些变量名相同的宏，如宏x、y、z，则编译很可能会出错，而且这种错误很难查找。

注意：编译UDF一定要单击本软件中的编译按钮来编译，假如使用VC++本身的编译工具，则编译很可能会出错。这时，要退出本软件，然后重新运行。

3. 关于程序调试

本书编写时该程序不支持在线调试，但是本书作者已经找到支持程序调试的方法。相信本书出版时，本软件已经可以支持在线调试。软件的更新请参考链接：

http://forum.cfluid.com/cgi-bin/LB5000/topic.cgi?forum=51&topic=435&show=0

第 9 章　动网格计算实例

9.1　悬浮生物载体在移动床运动的模拟

9.1.1　问题描述

如图 9-1 所示，悬浮生物载体移动床内除悬浮生物载体外，是空气和污水混合的气液两相流，气液两相流采用 Eularian 模型描述，悬浮载体假想为圆球形。

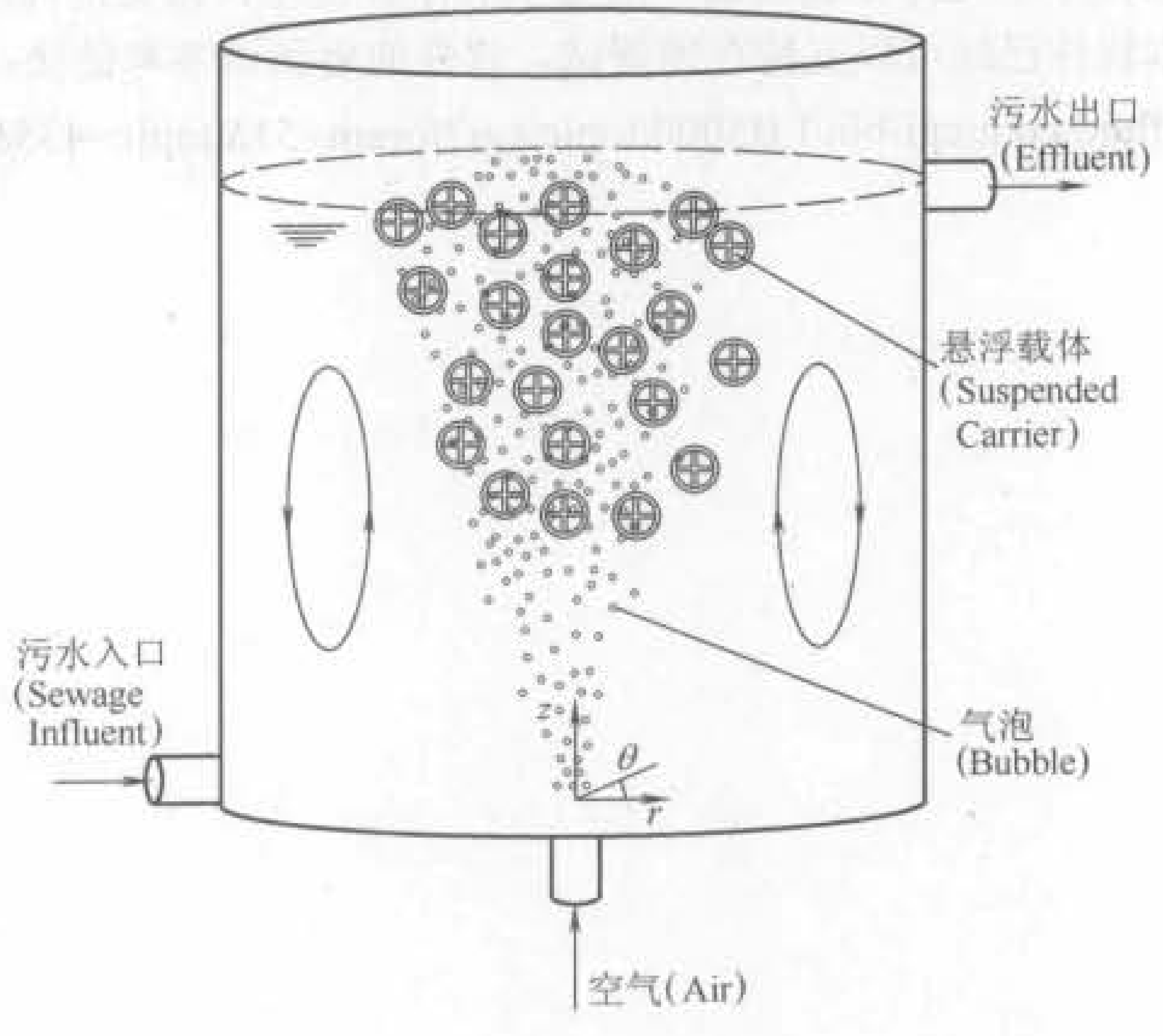


图 9-1　悬浮生物载体移动床内流场物理模型

空气和污水的进口采用速度边界条件，出水口采用压力出口边界，上部为气液接触的自由液面，最上面是气体压力出口，便于空气离开液体后自由逸出，其他边界为无滑移壁面边界。计算中所需物性参数与边界条件如表 9-1 所示。

表 9-1　计算的物性参数及边界条件

相	物性参数		边界条件			
	密度/kg·m^{3}	动力粘度/Pa·s	空气入口	污水入口	污水出口	空气出口
空气	1.225	1.789×10^{-5}	D=3mm	D=5mm	D=3mm	D=120mm
污水	1000.78	1.007×10^{-3}	v=0.708m/s	v=0.0064m/s		

考虑到计算方便，将上述问题简化为通过各进出管的二维问题。

9.1.2　具体计算过程

1. 几何建模

在 AutoCAD 中进行二维计算区域建模，如图 9-2 所示，长 120mm，高 205mm，载体圆直径 20mm。注意要在水进口、气进口、溢流出口留有断点，以便在 Gambit 中定义边界条件。采用 region 命令将计算区域做成两个区域，而后输出成 dm_1.sat 文件供 Gambit 使用。

2. 网格划分

在 Gambit 中，导入上述 dm_1.sat 文件，首先采用面的差运算，将两个区域合成一个区域，考虑到悬浮载体是自由运动，计算区域的网格将在计算中发生大变形，所以采用三角形网格进行计算区域的划分，即在 Mesh Faces 面板上的设置为 Elements：Tri，Type：Pave，Interval Size：2，其他保持默认值，整个网格如图 9-3 所示，共 7138 个节点，13918 个单元。

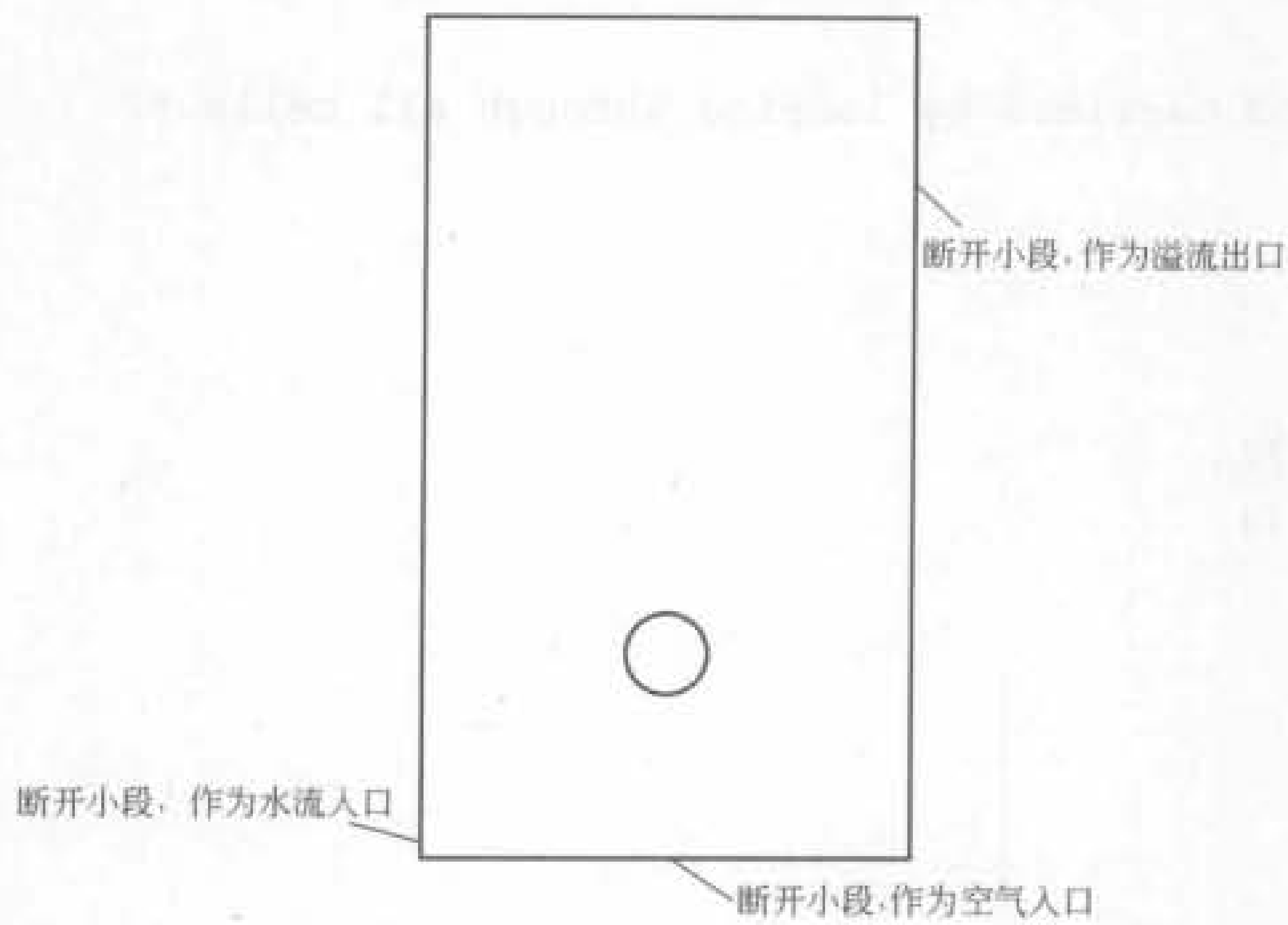


图 9-2　计算区域模型

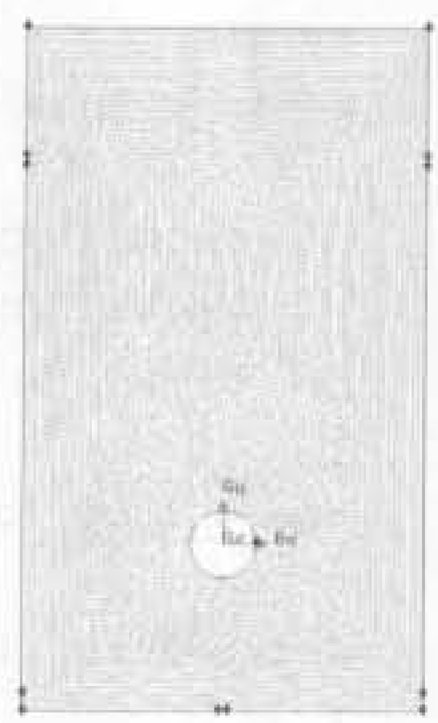


图 9-3　计算网格图

在 Specify Boundary Types 面板中设置边界条件，左下角的小段设置为速度入口边界(VELOCITY_INLET)，命名为 waterinlet；正下方的小段设置为速度入口(VELOCITY_INLET)，命名为 airinlet；右上部的小段设置为压力出口边界(PRESSURE_OUTLET)；圆的边线设置为墙边界(WALL)，命名为 movingwall。

设置完成后保存，输出为 dm_1.msh 文件。

3. UDF 编写与编译

采用自主开发的 VC++编译平台(FLUENT VC++6.0 Programmer)进行悬浮载体运动 UDF 的编写与编译，不考虑悬浮载体的旋转，根据悬浮生物载体周围的液流速度来确定悬浮载体在移动床中的运动。具体程序如下：

```
#include "stdafx.h"
#include "afxtempl.h"
#include "udf.h"
#include "memory.h"
extern "C"{
DEFINE_CG_MOTION(movewall, dt, vel, omega, time, dtime)
{
   Thread *t;//cell thread in dynamic boundary
   cell_t c;
   real cx,cy;
   CArray<real,real&> vx,vy; //Carry variable
   int i;
   int n=0;

   if (!Data_Valid_P ())
   return;
   /* get the thread pointer for which this motion is defined */
   t = DT_THREAD (dt);
   /* get velocity around carriers by looping through all cells */
   cx=0;
   cy=0;
   begin_c_loop (c, t)
   {
       vx.Add (C_U(c,t));
       vy.Add (C_V(c,t));
       n++;
   }
   end_c_loop (c, t)

   for(i=0;i<n;i++)
   {
       cx=cx+vx[i];
       cy=cy+vy[i];
   }
   cx=cx/n;
   cy=cy/n;
   vel[0]=cx;
   vel[1]=cy;
   Message ("\n n=%d, vx = %f,  vy =%f\n",n,vel[0],vel[1]);
}
}
```

编写好上述 UDF，单击如图 9-4 所示的 FLUENT VC++6.0 Programer 2.0 中的 Compiling 按钮直接进行编译，VC 编译窗口出现 0 个 errors 时，表示编译成功，该 UDF 可以导入 Fluent 使用了。

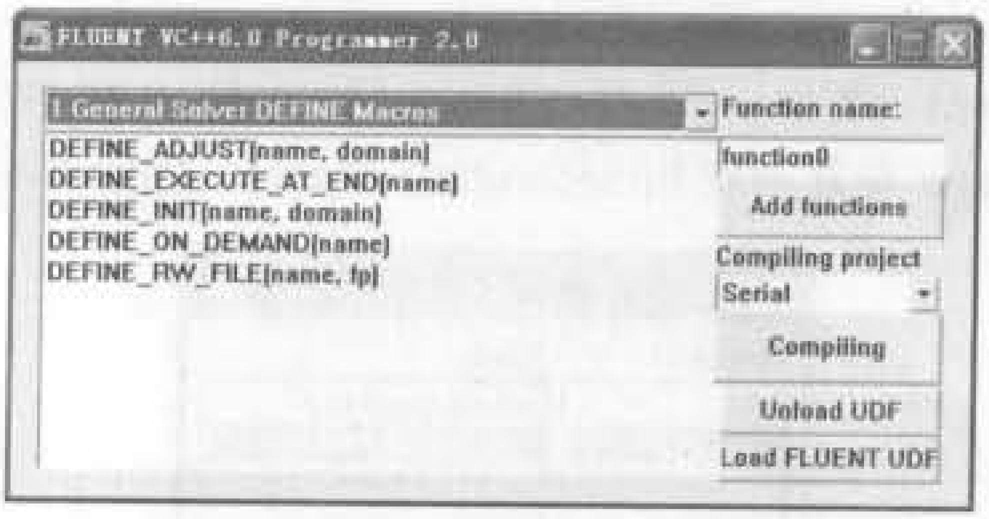


图 9-4　FLUENT VC++6.0 Programmer 2.0 界面

4. Fluent 计算设置

1)　读入.msh 文件

选择 File→Read→Case 命令，读入 Gambit 输出的 dm_1.msh 文件，并检查网格(选择 Grid→Check 命令)，注意最小体积要大于 0。选择 Grid→Smooth/Swap 命令，消除网格的扭曲。

选择 Grid→Scale 命令，将计算区域的单位设置为 mm。

2)　模型设置

选择 Define→Models→Solver 命令，打开 Solver 对话框(可参阅图 2-12)，因为悬浮载体运动属于瞬态计算，在 Time 选项组中选中 Unsteady 单选按钮，其他保持默认值。

选择 Define→Models→Multiphase 命令，打开 Multiphase Model 对话框，选择 Eulerian 多相流模型，Number of Phases 设置为 2。

选择 Define→Models→Viscous 命令，打开 Viscous Model 对话框，选择 k-epsilon 模型，其余保持默认值。

3)　材料设置

选择 Define→Materials 命令，打开 Materials 对话框，如图 9-5 所示，创建 sewage 材料。

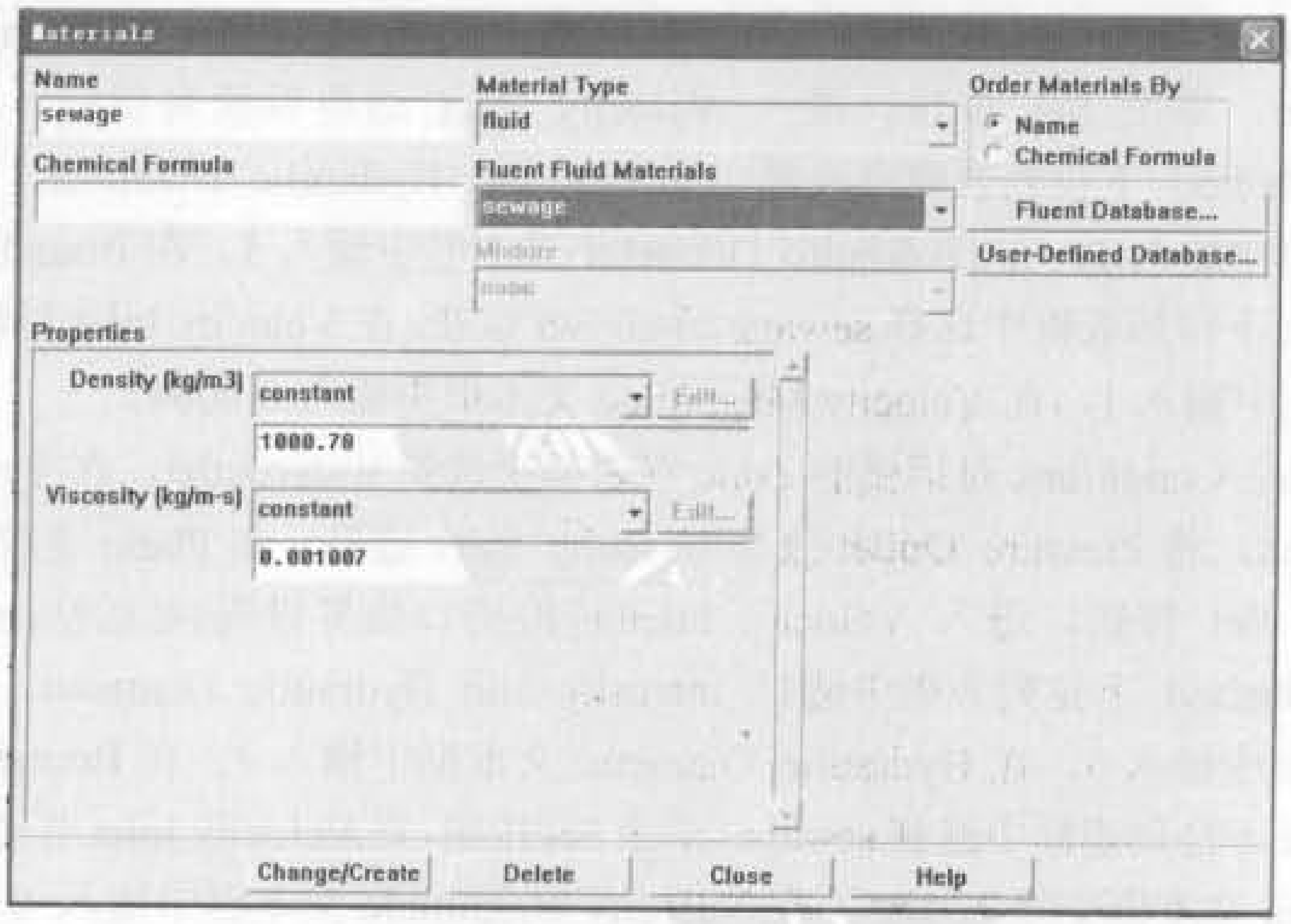


图 9-5　创建 sewage 材料

4) 相设置

选择 Define→Phases 命令，打开 Phases 对话框，将较轻相空气设置为主相(primary-phase)，将 sewage 设置为次相(secondary-phase)，如图 9-6 所示。

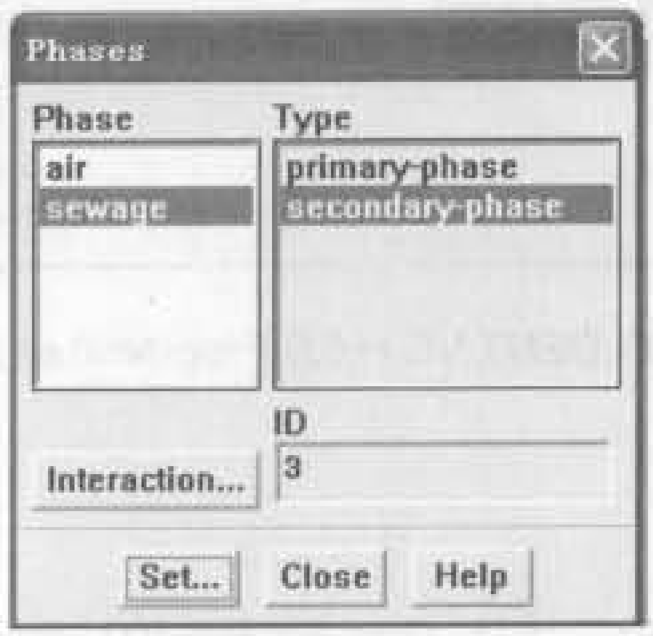


图 9-6 相设置

5) 操作条件设置

选择 Define→Operating Conditions 命令，打开 Operating Conditions 对话框，设置参考压力的位置为 x=0，y=150，选中 Gravity，设置 y 方向的重力加速度为-9.81m/s^2。

6) 边界条件设置

选择 Define→Boundary Conditions 命令，打开 Boundary Conditions 对话框(可参阅图 2-18 和图 5-14)。

在 Zone 列表框中选择 airinlet，在 Phase 下拉列表框中选择 mixture，单击 Set 按钮，进入 Velocity Inlet(速度进口边界设置)对话框；在 Turbulence Specification Method 下拉列表框中选择 Intensity and Hydraulic Diameter，在 Turbulence Intensity 文本框中输入 5，在 Hydraulic Diameter 文本框中输入 3。在 Boundary Conditions 对话框的 Phase 下拉列表框中选择 air，单击 Set 按钮，在 Velocity Inlet 对话框的 Velocity Magnitude 文本框中输入 0.708。

在 Boundary Conditions 对话框的 Zone 列表框中选择 waterinlet，在 Phase 下拉列表框中选择 mixture，单击 Set 按钮，进入 Velocity Inlet 边界设置对话框，在 Turbulence Specification Method 下拉列表框中选择 Intensity and Hydraulic Diameter，在 Turbulence Intensity 文本框中输入 5，在 Hydraulic Diameter 文本框中输入 5。在 Boundary Conditions 对话框的 Phase 下拉列表框中选择 sewage，单击 Set 按钮，在 Velocity Inlet 对话框的 Volume Fraction 文本框中输入 1，在 Velocity Magnitude 文本框中输入 0.0064。

在 Boundary Conditions 对话框的 Zone 列表框中选择 wateroutlet，在 Type 列表框中单击 Velocity Inlet，将 Pressure Outlet 改为 Velocity Inlet 边界，在 Phase 下拉列表框中选择 mixture，单击 Set 按钮，进入 Velocity Inlet(速度进口边界设置)对话框；在 Turbulence Specification Method 下拉列表框中选择 Intensity and Hydraulic Diameter，在 Turbulence Intensity 文本框中输入 5，在 Hydraulic Diameter 文本框中输入 3。在 Boundary Conditions 对话框的 Phase 下拉列表框中选择 sewage，单击 Set 按钮，在 Velocity Inlet 对话框的 Backflow Volume Fraction 文本框中输入 0.91，在 Velocity Magnitude 文本框中输入-0.01777。

在 Boundary Conditions 对话框的 Zone 列表框中选择 airoutlet，在 Phase 下拉列表框中选择 mixture，单击 Set 按钮，进入 Pressure Inlet(压力进口边界设置)对话框，在 Turbulence

Specification Method 下拉列表框中选择 Intensity and Hydraulic Diameter，在 Turbulence Intensity 文本框中输入 5，在 Hydraulic Diameter 文本框中输入 120，其余保持默认值。

7)　动网格设置

选择 Define→User-Defined→Functions→Manage 命令，打开 UDF Library Manager 对话框，如图 9-7 所示，在 Library Name 中输入 c:\dmmove1\libudf，单击 Load 按钮，在 UDF Libraries 中出现 c:\dmmove1\libudf，表示 UDF 导入成功(也可以直接通过这个编译器的 Load Fluent UDF 按钮加载函数)。

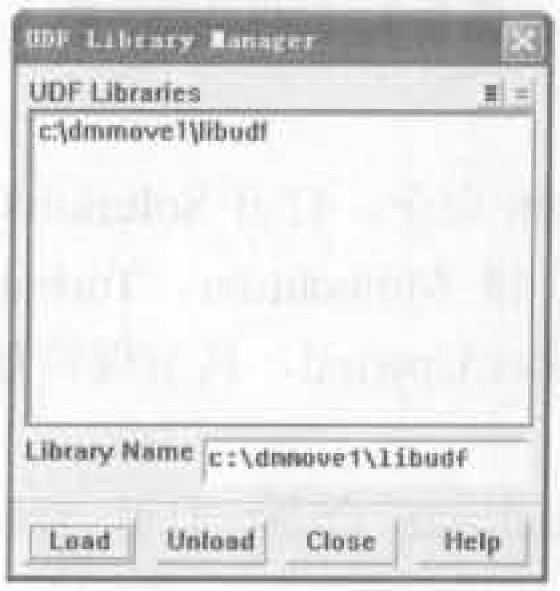


图 9-7　UDF 导入对话框

选择 Define→Dynamic Mesh→Parameters 命令，打开 Dynamic Mesh Parameters 对话框。

在 Models 选项组中选中 Dynamic Mesh 复选框，对话框扩展，在 Mesh Methods 选项组中选中 Smoothing 和 Remeshing 复选框，单击右上边的 Remeshing 标签，切换到 Remeshing 选项卡进行参数设置；在 Options 选项组中选中 Size Function 和 Must Improve Skewness 复选框，如图 9-8 所示，单击 Mesh Scale Info 按钮，出现如图 9-9 所示的 Mesh Scale Info 对话框，上面的值可以作为 Dynamic Mesh Parameters 对话框上相应位置填写的参考。

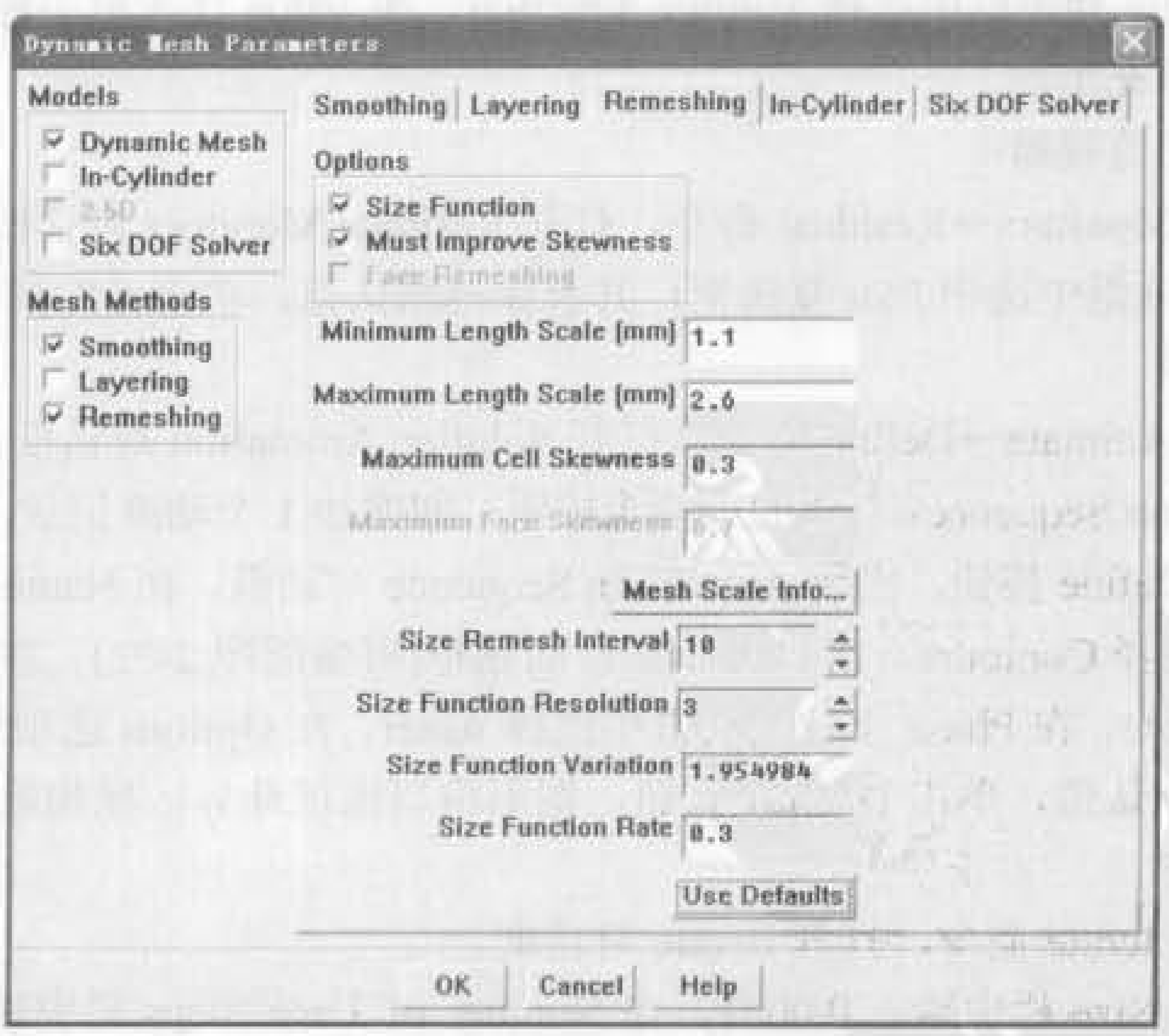


图 9-8　Remeshing 参数设置

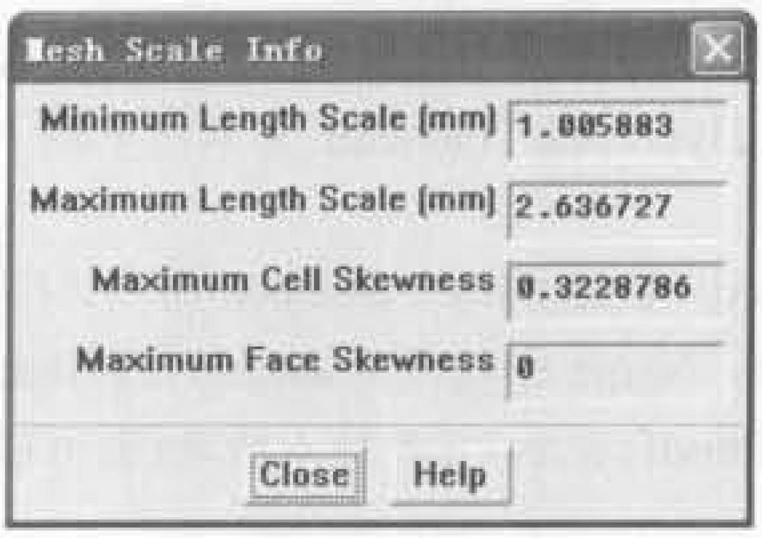


图 9-9 网格参考信息

8) 求解参数设置

选择 Solve→Controls→Solution 命令，打开 Solution Controls 对话框(可参阅图 6-12)。

在 Discretization 选项组中，将 Momentum、Turbulence Kinetic Energy、Turbulence Dissipation Rate 设定为 Second Order Upwind，其余保持默认值。

9) 流场初始化

选择 Solve→Initializate→Initializate 命令，打开 Solution Initialization 对话框(可参阅图 6-13)。

在 Turbulence Kinetic Energy 文本框中输入 6.703989e-05，在 Turbulence Dissipation Rate 文本框中输入 0.001791256，其余设为 0。

选择 Adapt→Region 命令，打开 Region Adaption 对话框。

在 X Min 文本框中输入-60，在 X Max 文本框中输入 60，在 Y Min 文本框中输入-50，在 Y Max 文本框中输入 120，单击 Mark 按钮，标记一个区域，单击 Close 按钮结束。

选择 Solve→Initializate→Patch 命令，打开 Patch 对话框(可参阅图 6-30)。

在 Registers to Patch 列表框中选择刚标记的 hexahedron-r0，在 Phase 下拉列表框中选择 sewage，在 Variable 列表框中选择 Volume Fraction，在 Value 文本框中输入 1，单击 Patch 按钮，为该区域预充填水。

10) 设置残差监视器

选择 Solve→Monitors→Residual 命令，打开 Residual Monitors 对话框(可参阅图 6-14)。

在 Opions 选项组中选中 Plot 复选框，其余保持默认值，单击 OK 按钮结束。

11) 动画记录

选择 Solve→Animate→Define 命令，打开 Solution Animation 对话框。

单击 Animation Sequences 右边的上箭头按钮，即增加 1 个动画记录，下面出现一亮显行；单击后面的 Define 按钮，出现 Animation Sequence 对话框，在 Name 中输入 dm_1，在 Display Type 中选择 Contours，出现 Contours 对话框(可参阅图 2-32)；在 Contours of 下拉列表框中选择 Phase，在 Phase 下拉列表框中选择 water，在 Options 选项组中选中 Filled 复选框，其余保持默认值，单击 Display 按钮，图形窗口出现计算区域初始的相分布。

12) 迭代计算

选择 Solve→Iterate 命令，打开 Iterate 对话框。

在 Time Step Size 栏中输入 0.0001，在 Number of Time Steps 栏中输入 3000，在 Max Iterations per Time Step 栏中输入 60，其余保持默认值。

选择 File→Write→Autosave 命令，打开 Autosave Case/Data 对话框，在 Autosave Case File

Frequency 和 Autosave Data File Frequency 栏中均输入 200，表示 200 步后自动保存一次 case 和 date 文件，单击 OK 按钮关闭该对话框。

单击 Iterate 对话框中的 Iterate 按钮，开始迭代计算。

5. 结果后处理

1) 云图显示

选择 Display→Contours 命令，打开 Contours 对话框，选中 Filled 复选框。

在 Contours of 下拉列表框中选择 Pressure，在 Phase 下拉列表框中选择 mixture，单击 Display 按钮，显示如图 9-10 所示的压力云图。

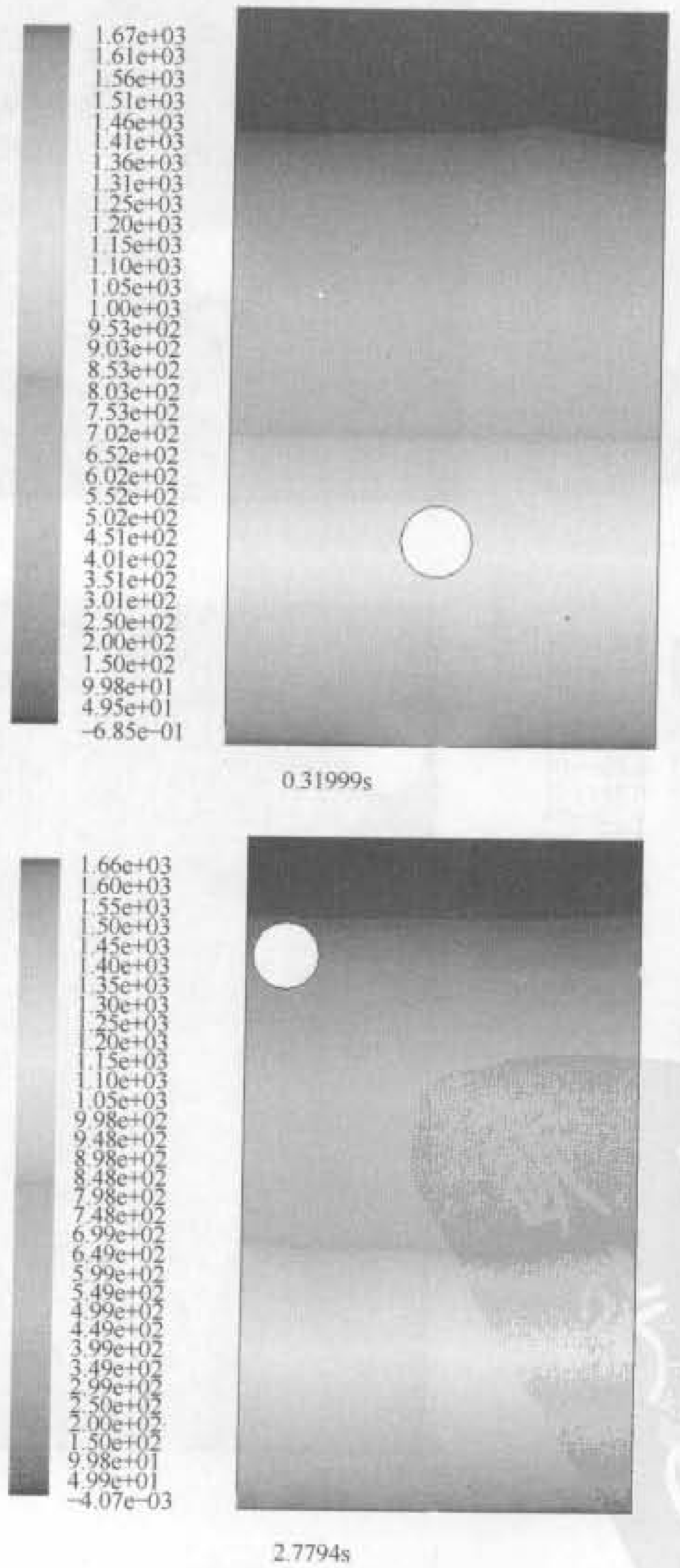


图 9-10 压力云图

在 Contours of 下拉列表框中选择 Velocity，在 Phase 下拉列表框中选择 mixture，单击 Display 按钮，显示如图 9-11 所示的速度云图。

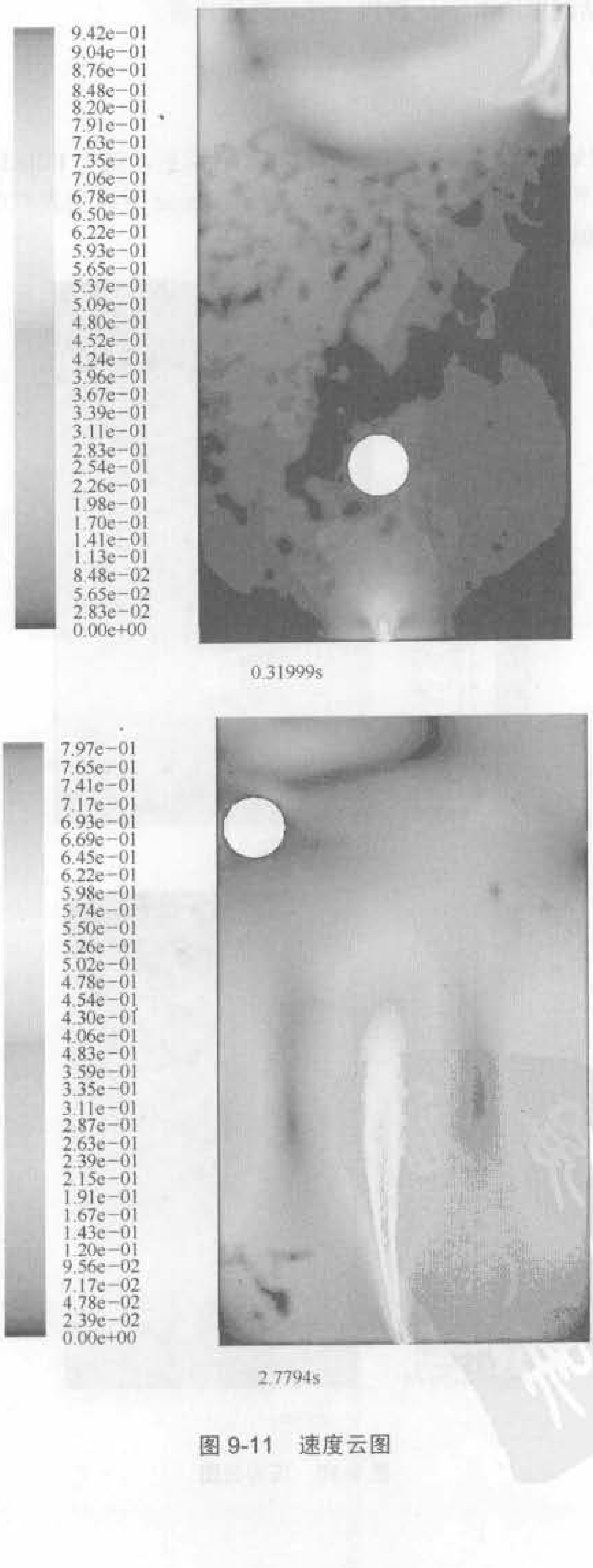


图 9-11 速度云图

在 Contours of 下拉列表框中选择 Phases，在 Phase 下拉列表框中选择 sediments，单击 Display 按钮，显示如图 9-12 所示的相分布云图。

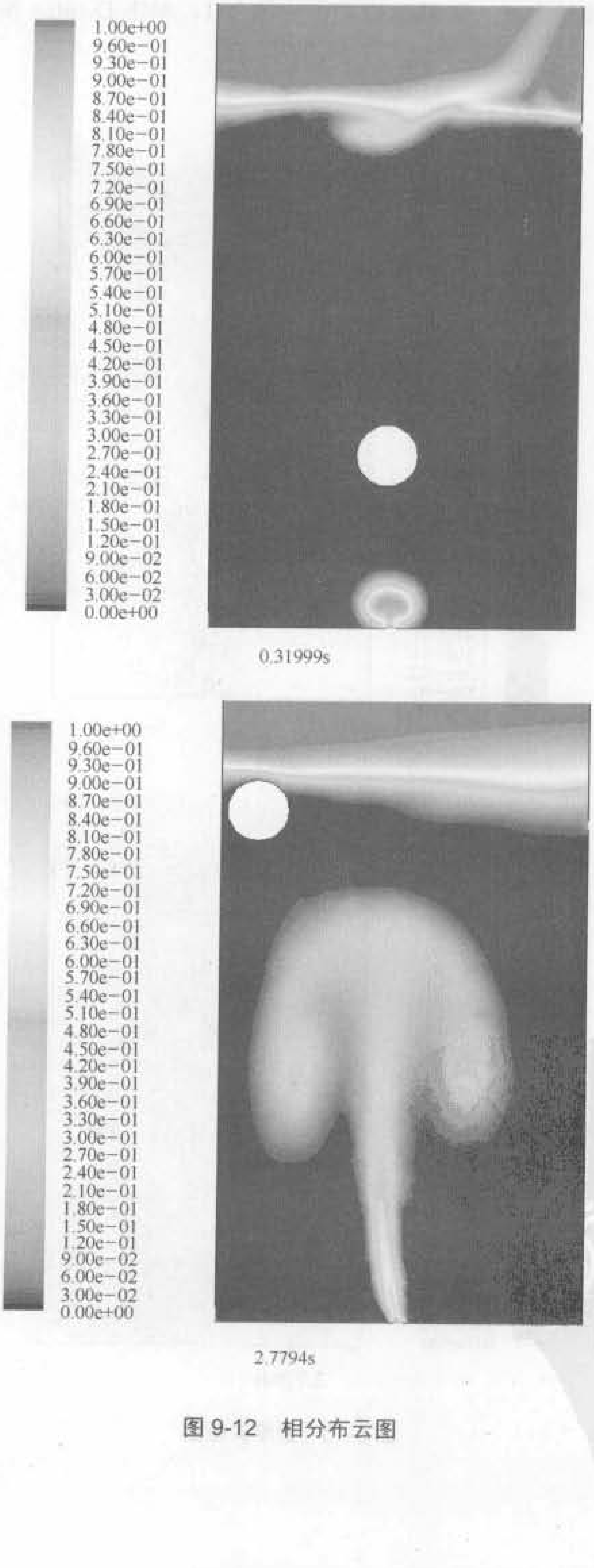


图 9-12　相分布云图

2) 矢量显示

选择 Display→Velocity Vectors 命令，打开 Vectors 对话框(可参阅图 2-36)。

在 Scale 文本框中输入 5，在 Skip 微调框中输入 1，单击 Display 按钮，显示如图 9-13 所示的速度矢量图。

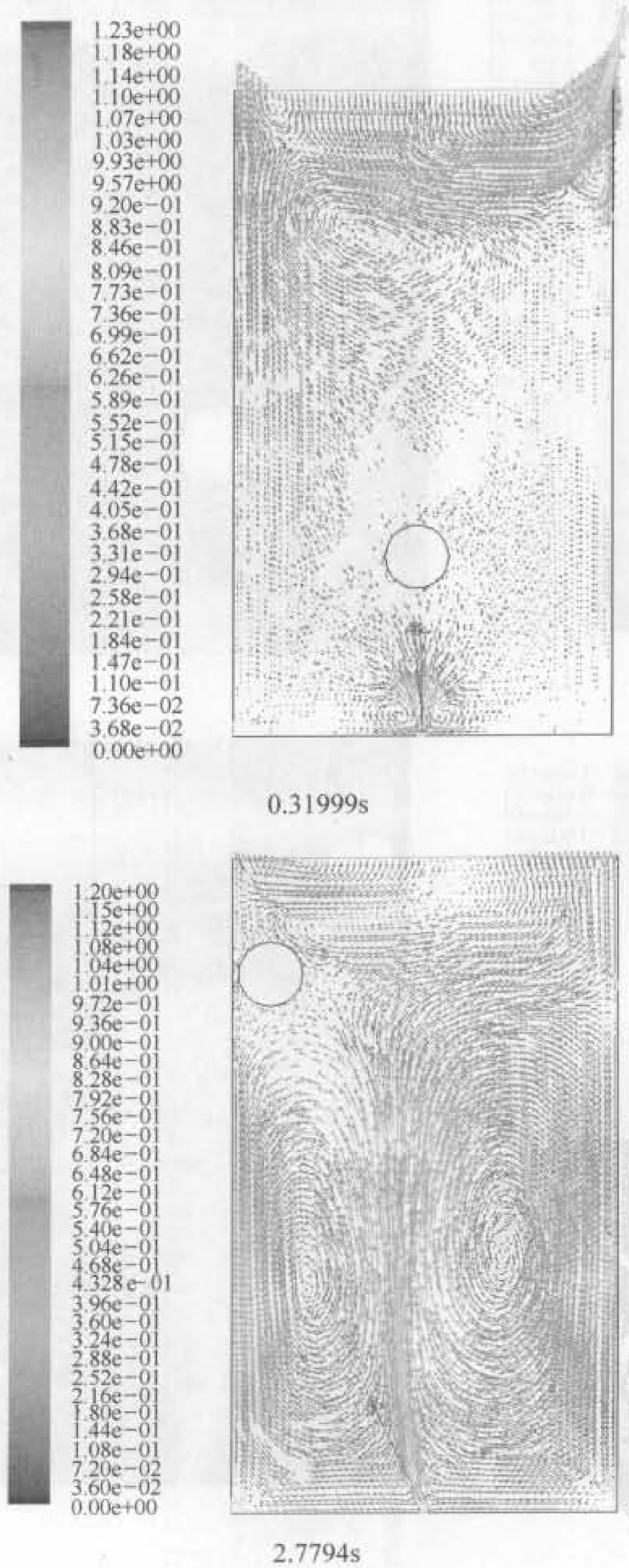


图 9-13 速度矢量图

3)　XY 曲线显示

显示计算区域任意线段或任意面上的物理量的变化情况。这里在 sediments 面上创建一根线段。

选择 Surface→Line/Rake 命令，打开 Line/Rake Surface 对话框。输入 x0=-20，x1=20，y 坐标值均输入 80，单击 Create 按钮，创建名为 line-7(该名称根据自己的情况命名，这里是默认名称)的线段。

选择 Plot→XY Plot 命令，打开 Solution XY Plot 对话框(可参阅图 2-38)。在 Y Axis Function 下拉列表框中选择 Phases，在 Phase 下拉列表框中选择 sewage，在 surfaces 列表框中选择 line-7。单击 Curves 按钮，显示如图 9-14 所示的 Curves-Solution XY Plot 对话框，在 Pattern 下拉列表框中，选择线，单击 Apply 按钮应用设置，再单击 Close 按钮关闭该对话框。在 Solution XY Plot 对话框中单击 Plot 按钮，显示如图 9-15 所示的计算区域直线 line-7 上的压力分布曲线图(图中横坐标代表位置，单位为 mm；纵坐标代表混合物静态压力，单位为 Pa)。

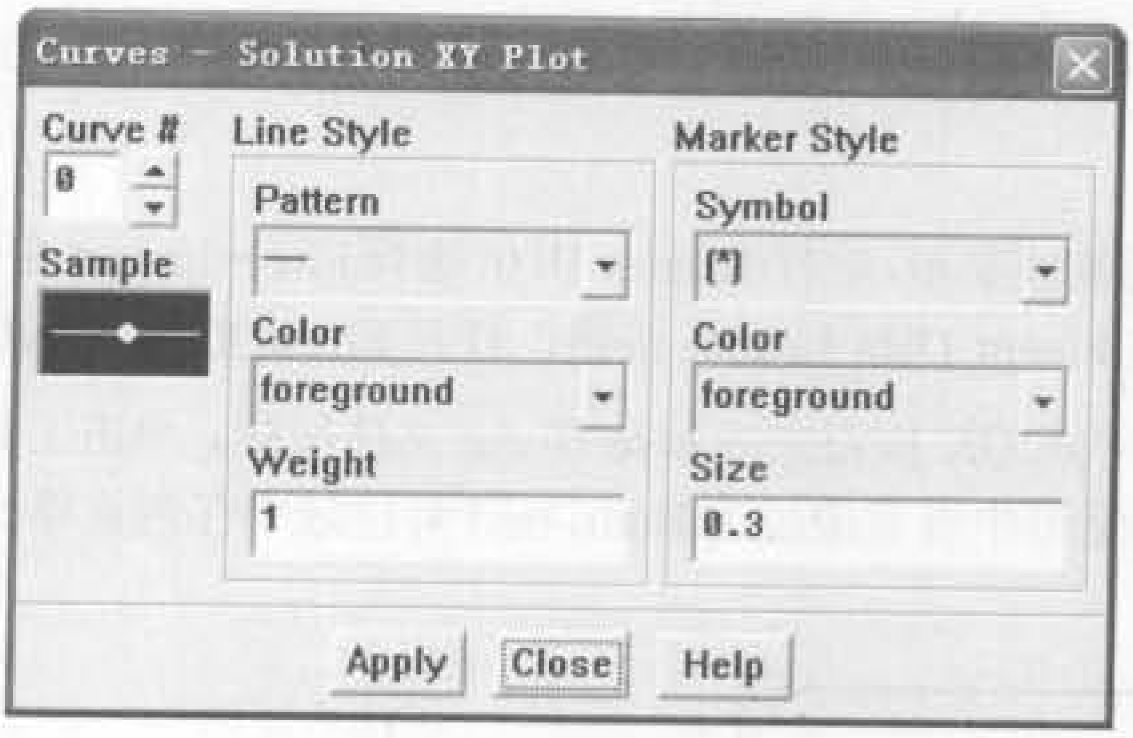


图 9-14　Curves-Solution XY Plot 对话框

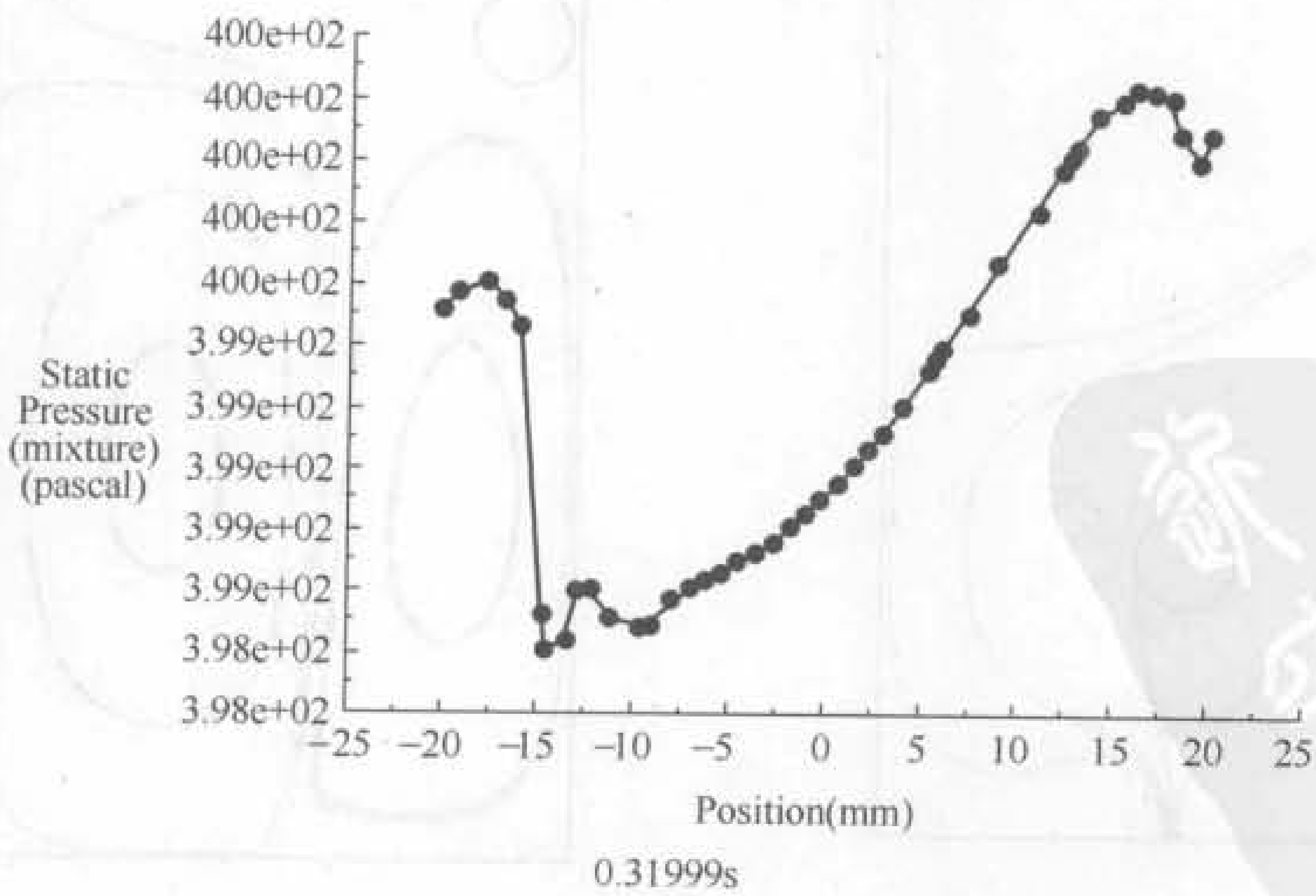


图 9-15　计算区域直线 line-7 上的压力分布曲线图

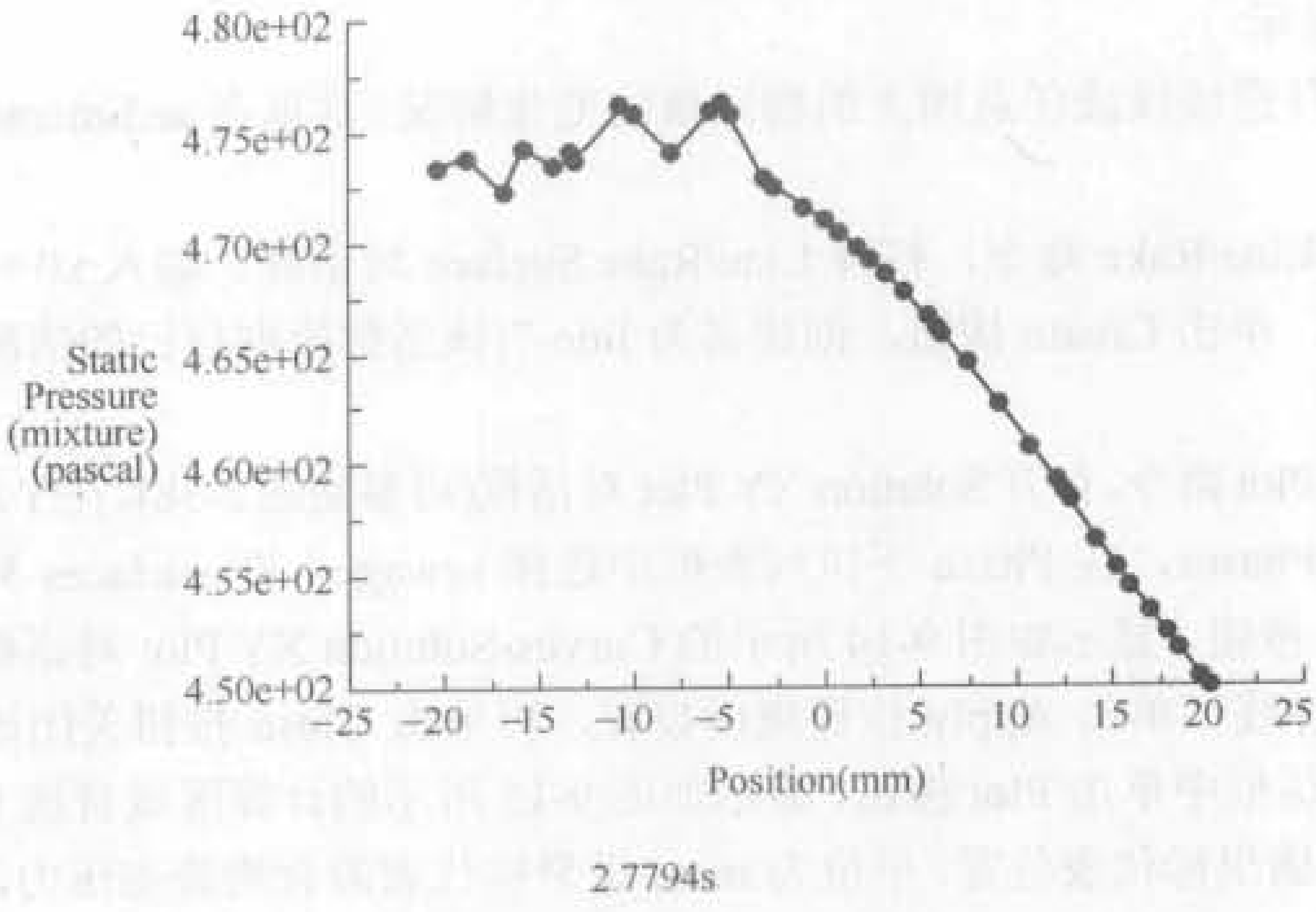


图 9-15 （续）

4) 流线显示

在 Tecplot 中进行流线显示，运行 Tecplot 10.0，选择 File→Import 命令，打开 Select Import Format 对话框；单击 Fluent Data Loader 按钮，打开 Fluent Data Loader 对话框；选择对应的 case 和 dat 文件，单击 OK 按钮完成 case 和 dat 文件加载。单击工具栏中的按钮，在 Streamtrace Details 对话框中设置流线为 Both，在计算区域中任意点单击，就会显示如图 9-16 所示的流线图。

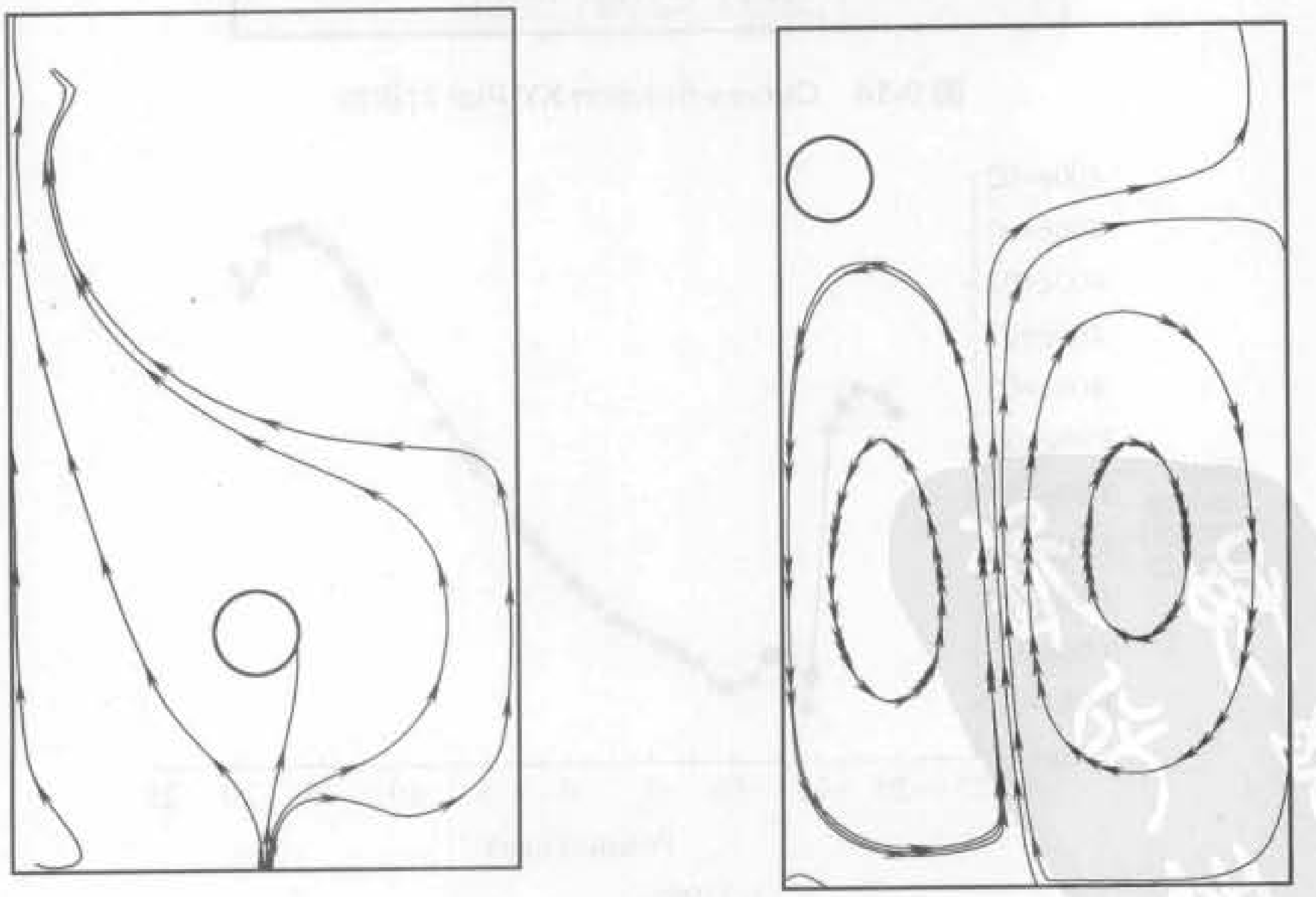


图 9-16 流线显示

9.2　齿轮泵的动态模拟

9.2.1　问题描述

由于齿轮泵内齿轮运动及工作介质流动的复杂性，其数值模拟工作比较复杂，国内对齿轮泵的模拟仅局限于流量模拟，没有进行详细的泵内流场模拟，国外对泵内流场模拟相对较多。Kris Riemslagh 等人采用拉格朗日-欧拉有限体积法计算了旋转置换泵(包括凸轮泵和齿轮泵)内流动，John Vande Voorde 等人利用 Fluent 软件计算了旋转容积泵的内部流场。齿轮泵内部动态模拟有助于真实地反映泵内流动的变化，而目前这方面的研究较少。本文采用动网格技术模拟柴油机润滑用齿轮泵的转动过程中的动态流动，为齿轮泵的结构优化及新型齿轮泵的设计提供参考。

齿轮泵的齿轮的模数为 2.75，齿数为 10，变位系数为 0.5，压力角为 20°，齿轮转速为 460r/min。计算流体的密度为 880kg/m^3，粘性系数为 0.04048N·s/m^2。

9.2.2　具体计算过程

1. 几何建模

在 CAXA 电子图板中绘制齿轮，该软件有绘制齿轮的命令，直接输入参数可以绘制所需齿轮，存储 dwg 文件，在 AutoCAD 中打开，复制齿轮构成齿轮对，两齿轮中心距要略大于标准啮合的中心距，以便于计算时网格的划分，并绘制外围的围边，结构如图 9-17 所示。要注意的是最好将坐标系原点移到齿轮的中心，这样以后确定转动中心时定位容易一些，定义好区域，输出 dm_2.sat 文件。

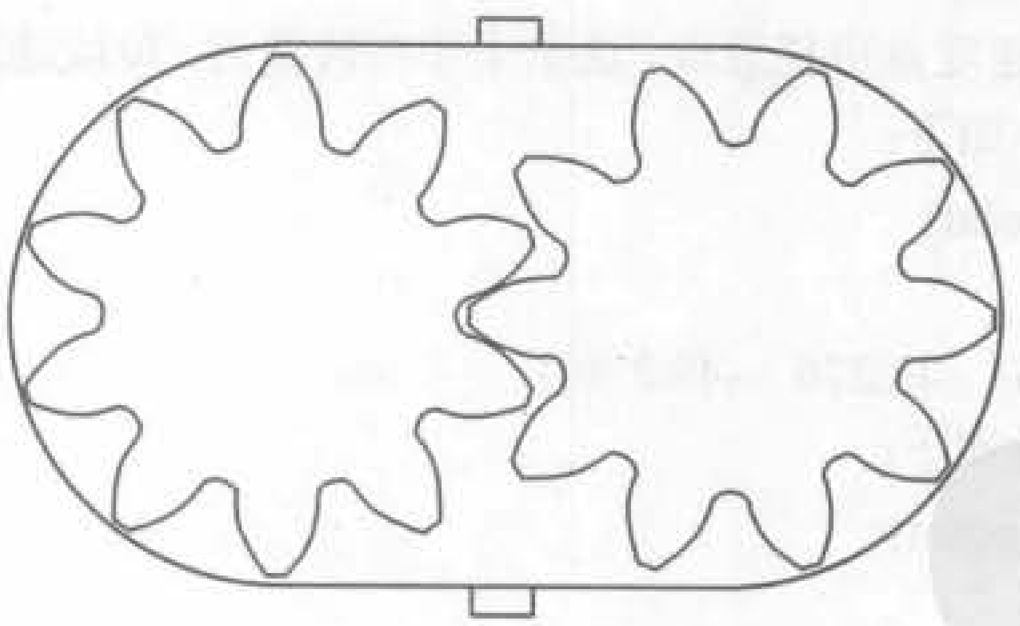


图 9-17　齿轮泵几何模型

2. 网格划分

在 Gambit 中，导入上述 dm_2.sat 文件，考虑到计算区域的网格将在计算中发生较大变形，所以采用三角形网格进行计算区域的划分，即在 Mesh Faces 对话框上的设置为 Elements：Tri，Type：Pave，Interval Size：0.2，其他保持默认值，整个网格如图 9-18 所示，共 21756 个节点，40988 个单元。

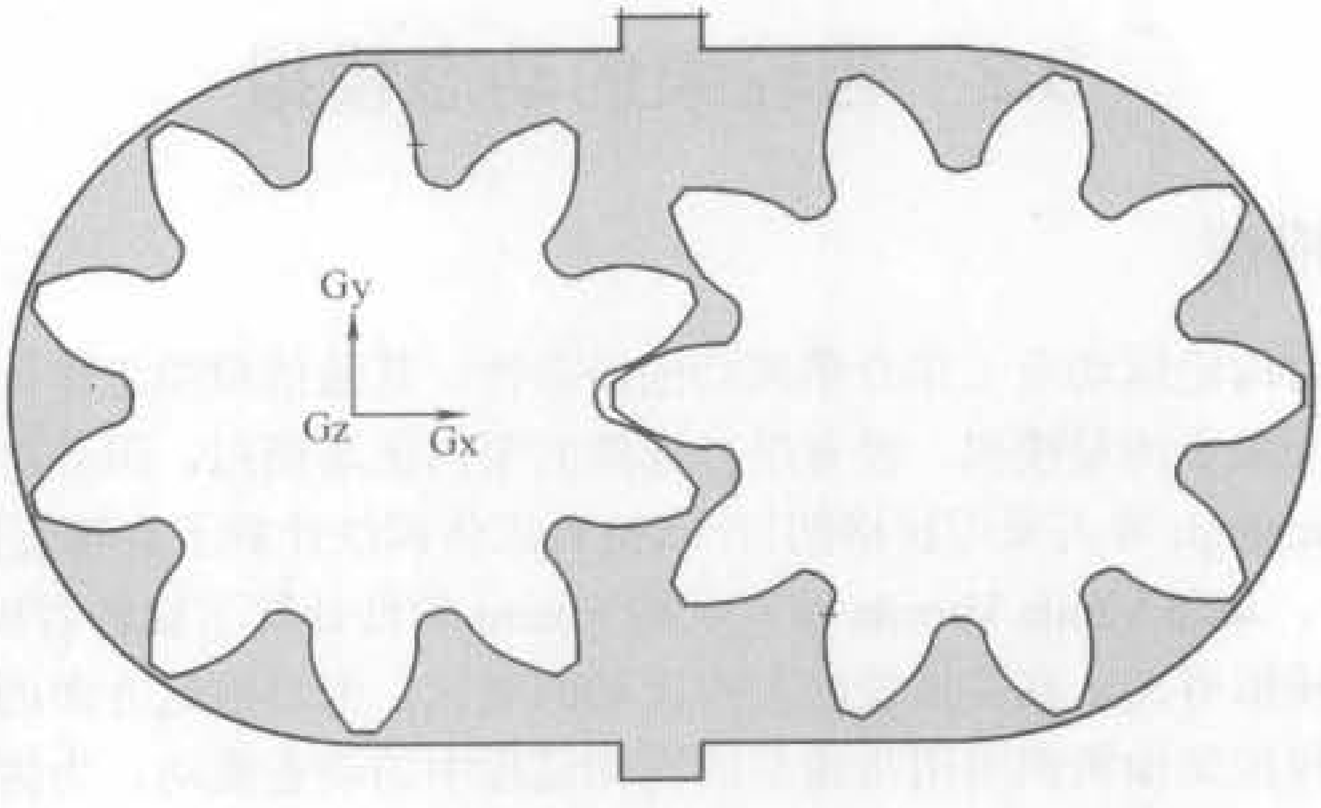


图 9-18 齿轮泵网格划分

在 Specify Boundary Types 面板中设置边界条件，最上边的线段设置为压力入口边界(PRESSURE_INLET)，命名为 oilin；最下边的线段设置为压力入口(PRESSURE _INLET)，命名为 oilout(本例是假定左边的齿轮逆时针旋转，故下边为压油口，上边为吸油口)；左边齿轮轮廓设置为墙边界(WALL)，命名为 movewall_left，同样，右边齿轮轮廓也设置为墙边界(WALL)，命名为 movewall_right。

设置完成后保存，输出为 dm_2.msh 文件。

3. 轮廓文件(或 UDF)编写

齿轮泵中齿轮旋转运动比较简单，就是绕轴中心转动，这里有两种方式实现：写轮廓文件和 UDF。下面分别介绍。

1) 轮廓文件(profile)编写

因为是恒速转动，速度点可任意取，这里采用三段速度，每段速度都一样，为 48.276r/s。

左边的 profile 文件如下：

((rotating_left 3 point)
(time 0 1 60)
(omega_z -48.276 -48.276 -48.276))

右边的 profile 文件如下：

((rotating_right 3 point)
(time 0 1 60)
(omega_z -48.276 -48.276 -48.276))

将上述 profile 文件分别编写，存储成文本文档 roting_left.txt 和 roting_right.txt，分别导入到 Fluent 中，就可以在相应的面板中看到该速度，7.3.2 节中有相关介绍。

2) UDF 编写与编译

采用自主开发的 VC++编译平台进行齿轮旋转运动 UDF 的编写与编译，本例只需一个旋转语句，omega[2]是绕 z 轴选择的变量，给定该值即可。具体程序如下：

```
#include "stdafx.h"
//write your include header here...
```

```
#include "udf.h"
extern "C"{
DEFINE_CG_MOTION(roting_left, dt, vel, omega, time, dtime)
{
omega[2]= 48.276;
}
DEFINE_CG_MOTION(roting_right, dt, vel, omega, time, dtime)
{
omega[2]=-48.276;
}
}
```

编写好上述 UDF，先在 VC++中编译，当错误只有 3 个时，可以单击 FLUENT VC++6.0 Programmer 2.0 中的 Compiling 按钮，出现 Compiling succeed，Please load the libudf to fluent 的提示，表示编译成功，该 UDF 可以导入 Fluent 使用了。

4. Fluent 计算设置

1)　读入 msh 文件

选择 File→Read→Case 命令，读入 Gambit 输出的 dm_2.msh 文件，并检查网格(Grid→Check)，注意最小体积要大于 0。选择 Grid→Smooth/Swap 命令，消除网格的扭曲。

选择 Grid→Scale 命令，将计算区域的单位设置为 mm。

2)　模型设置

选择 Define→Models→Solver 命令，打开 Solver 对话框，因为齿轮转动，属于瞬态计算，在 Time 选项组中选中 Unsteady 单选按钮，其他保持默认值。

选择 Define→Models→Viscous 命令，打开 Viscous Model 对话框，选择 k-epsilon 模型，其余保持默认值。

3)　材料设置

选择 Define→Materials 命令，打开 Materials 对话框，如图 9-19 所示，创建 oil 材料，单击 Change/Create 按钮后，单击 Yes 覆盖原有材料。

4)　操作条件设置

选择 Define→Operating Conditions 命令，打开 Operating Conditions 对话框，设置参考压力的位置为 x=15，y=20。

5)　边界条件设置

选择 Define→Boundary Conditions 命令，打开 Boundary Conditions 对话框。

在 Zone 列表框中选择 oilin，单击 Set 按钮，进入 Pressure Inlet(压力进口边界设置)对话框，在 Turbulence Specification Method 下拉列表框中选择 Intensity and Hydraulic Diameter，在 Turbulence Intensity 文本框中输入 5，在 Hydraulic Diameter 文本框中输入 4。

在 Boundary Conditions 对话框的 Zone 列表框中选择 oilout，单击 Set 按钮，进入 Pressure Outlet(压力出口边界设置)对话框，在 Turbulence Specification Method 下拉列表框中选择 Intensity and Hydraulic Diameter，在 Turbulence Intensity 文本框中输入 5，在 Hydraulic Diameter 文本框中输入 4。

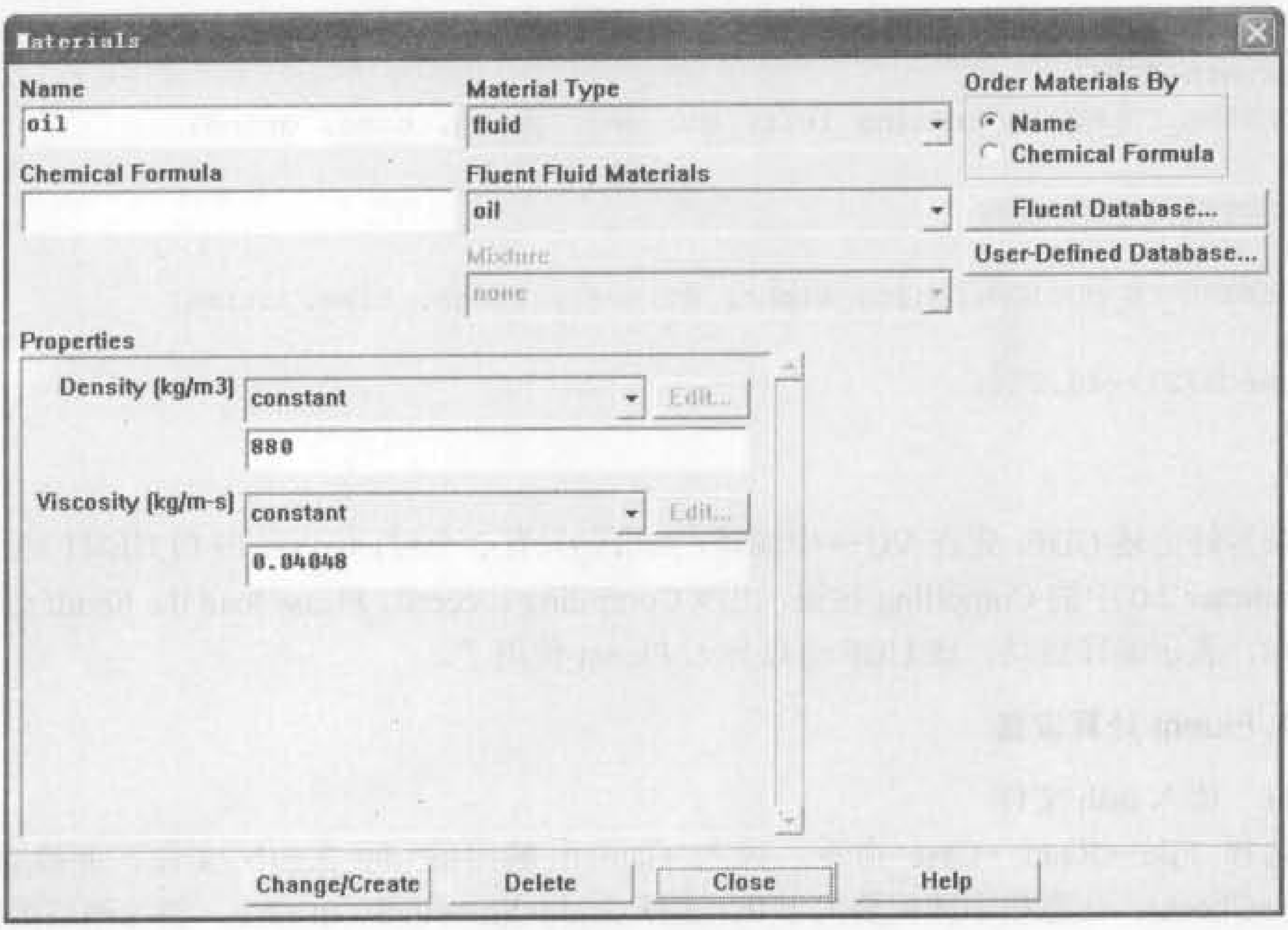


图 9-19 oil 材料创建

6) 动网格设置

选择 Define→User-Defined→Functions→Manage 命令，打开 UDF Library Manager 对话框，在 Library Name 文本框中输入 c:\dmmove2\libudf，单击 Load 按钮，在 UDF Libraries 列表框中出现 c:\dmmove2\libudf，表示 UDF 导入成功。

选择 Define→Dynamic Mesh→Parameters 命令，打开 Dynamic Mesh Parameters 对话框(可参阅图 7-6 和图 9-8)。在 Models 选项组中选中 Dynamic Mesh 复选框，对话框扩展，在 Mesh Methods 选项组中选中 Smoothing 和 Remeshing 复选框，单击右上边的 Remeshing 标签，切换到 Remeshing 选项卡进行参数设置；在 Options 选项组中选中 Size Function 和 Must Improve Skewness 复选框，单击 Mesh Scale Info 按钮，出现 Mesh Scale Info 对话框，上面的相应值可以作为 Dynamic Mesh Parameters 对话框中相应位置填写的参考。

7) 求解参数设置

选择 Solve→Controls→Solution 命令，打开 Solution Controls 对话框。

在 Discretization 选项组中，将 Momentum、Turbulence Kinetic Energy、Turbulence Dissipation Rate 设置为 Second Order Upwind，Pressure 设置为 PRESTO!，在 Pressure-Velocity Coupling 下拉列表框中选择 PISO，其余保持默认值，如图 9-20 所示。

8) 流场初始化

选择 Solve→Initialize→Initializate 命令，打开 Solution Initialization 对话框。

在 Compute From 下拉列表框中选择 all-zones，在 Turbulence Kinetic Energy 文本框中输入 0.01，在 Turbulence Dissipation Rate 文本框中输入 0.01，其余设为 0，单击 Init 按钮。

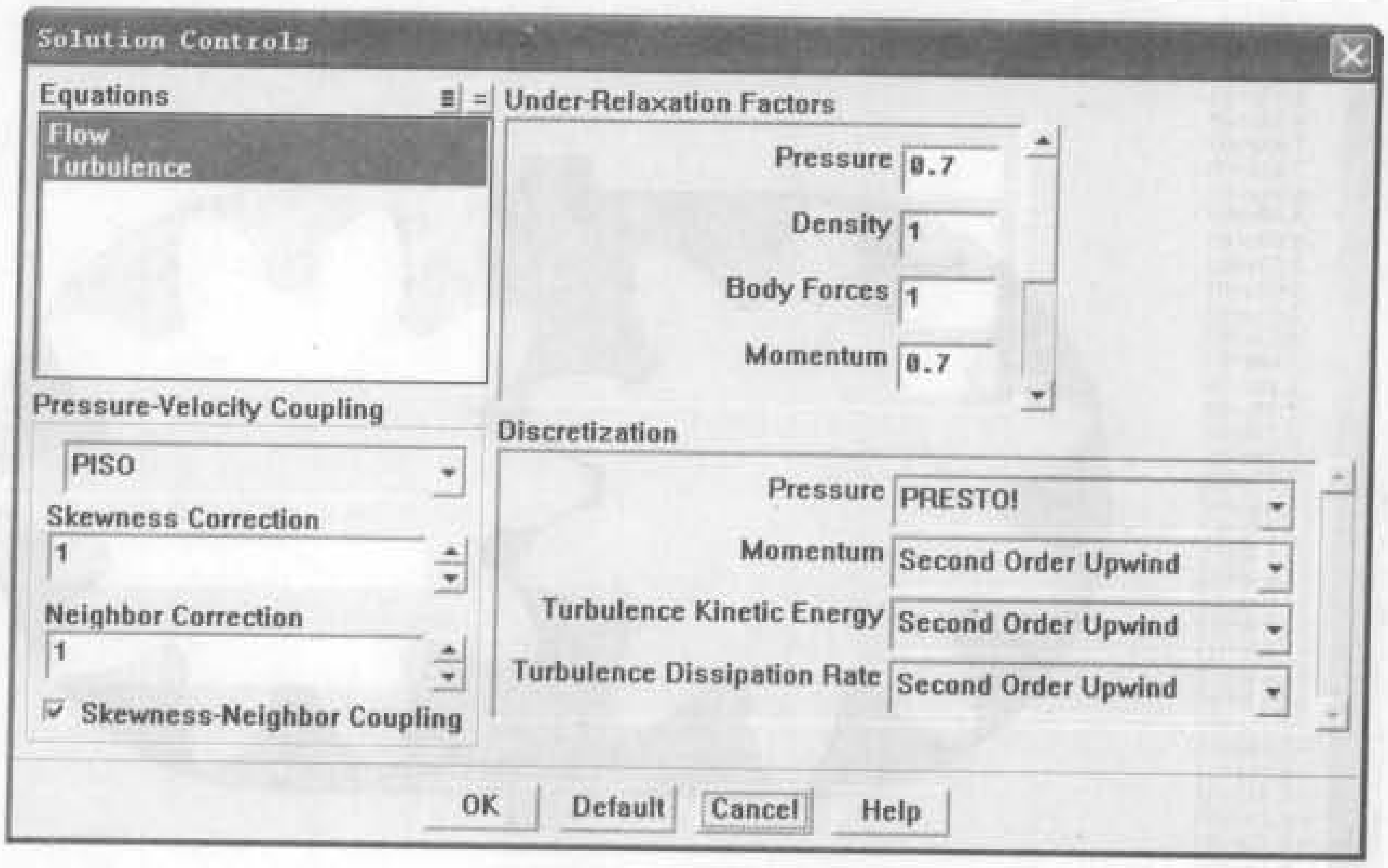


图 9-20　求解设置

9)　设置残差监视器

选择 Solve→Monitors→Residual 命令，打开 Residual Monitors 对话框。

在 Opions 选项组中选中 Plot 单选按钮，其余保持默认值，单击 OK 按钮结束。

10)　动画记录

选择 Solve→Animate→Define 命令，打开 Solution Animation 对话框。

单击 Animation Sequences 右边的上箭头按钮，即增加 1 个动画记录，下面出现一亮显行；单击后面的 Define 按钮，出现 Animation Sequence 对话框，在 Name 栏中输入 dm_2，在 Display Type 栏中选择 Contours，出现 Contours 对话框；在 Contours of 下拉列表框中选择 Pressure，在 Options 选项组中选中 Filled 复选框，其余保持默认值，单击 Display 按钮，图形窗口出现计算区域初始的压力分布。

11)　迭代计算

选择 Solve→Iterate 命令，打开 Iterate 对话框。

在 Time Step Size 栏中输入 0.0001，在 Number of Time Steps 栏中输入 3000，在 Max Iterations per Time Step 栏中输入 60，其余保持默认值。

选择 File→Write→Autosave 命令，打开 Autosave Cas/Data 对话框，在 Autosave Case File Frequency 和 Autosave Case File Frequency 栏中均输入 200，表示 200 步后自动保存一次 case 和 dat 文件，单击 OK 按钮关闭该对话框。

单击 Iterate 对话框中的 Iterate 按钮，开始迭代计算。

5. 结果后处理

1)　云图显示

选择 Display→Contours 命令，打开 Contours 对话框，选中 Filled 复选框。

在 Contours of 下拉列表框中选择 Pressure，在 Phase 下拉列表框中选择 mixture，单击 Display 按钮，显示如图 9-21 所示的压力云图。

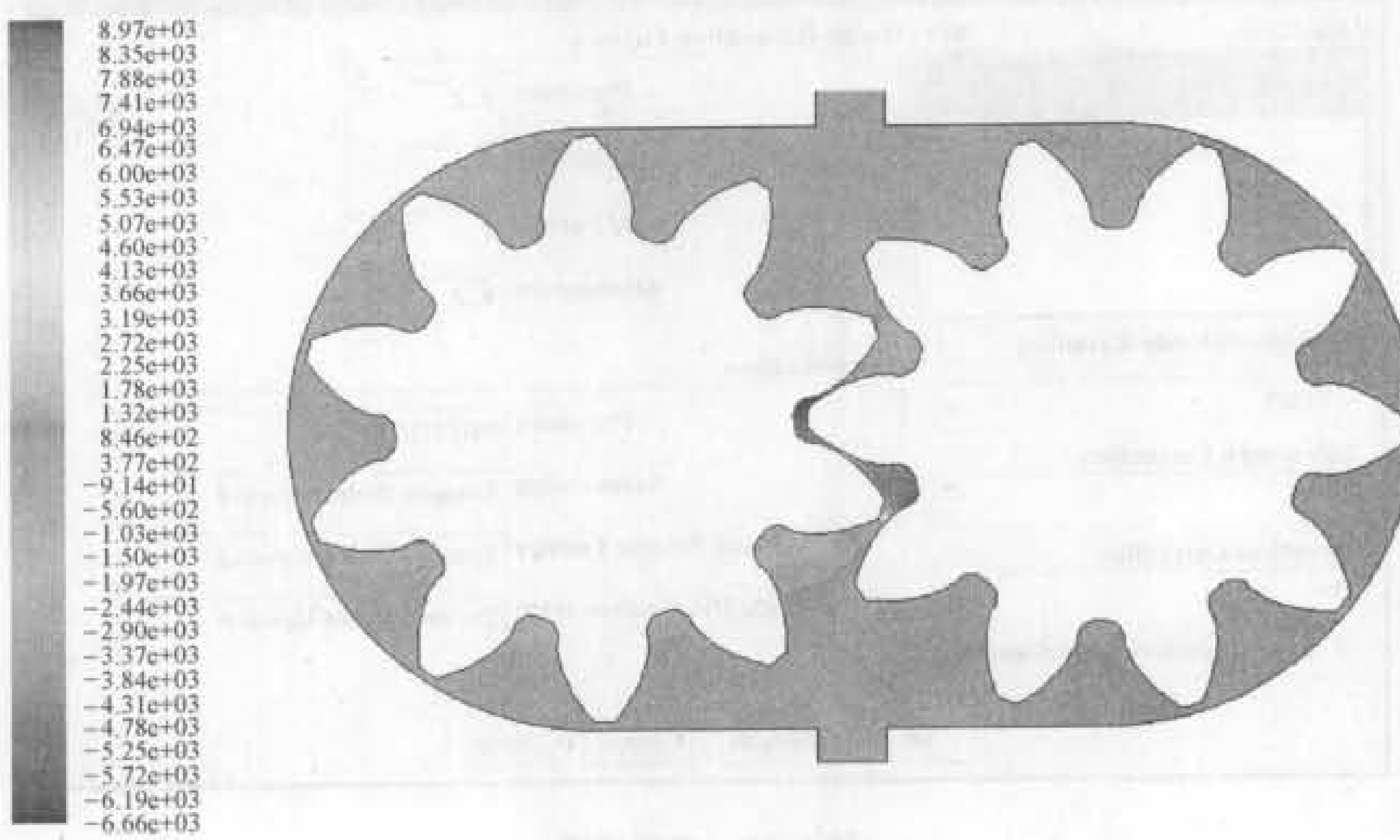


图 9-21 压力云图

在 Contours of 中选择 Velocity，单击 Display 按钮，显示如图 9-22 所示的速度云图。

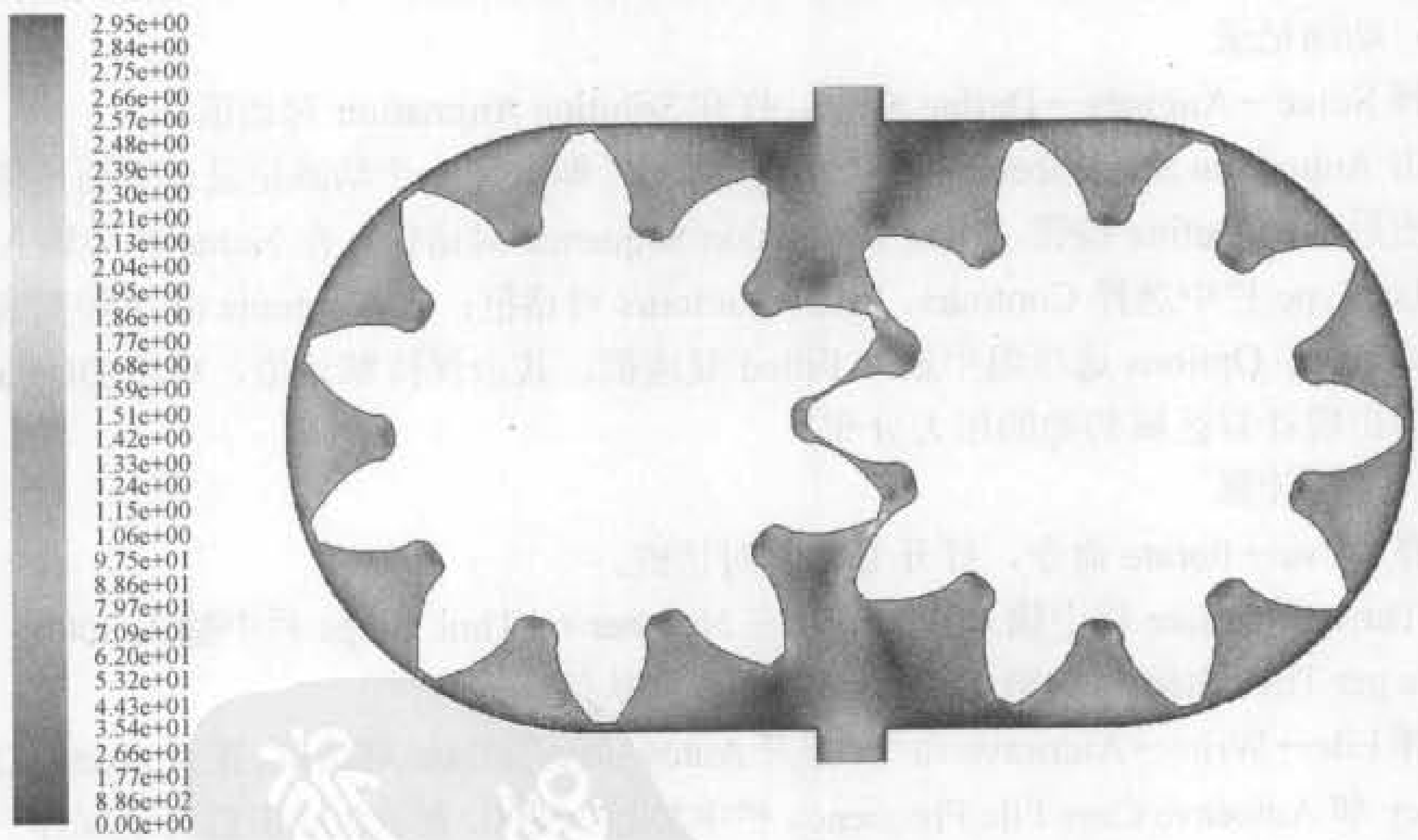


图 9-22 速度云图

2) 矢量显示

选择 Display→Velocity Vectors 命令，打开 Vectors 对话框。

在 Scale 文本框中输入 16，在 Skip 微调框中输入 4，单击 Display 按钮，显示如图 9-23 所示的速度矢量图。

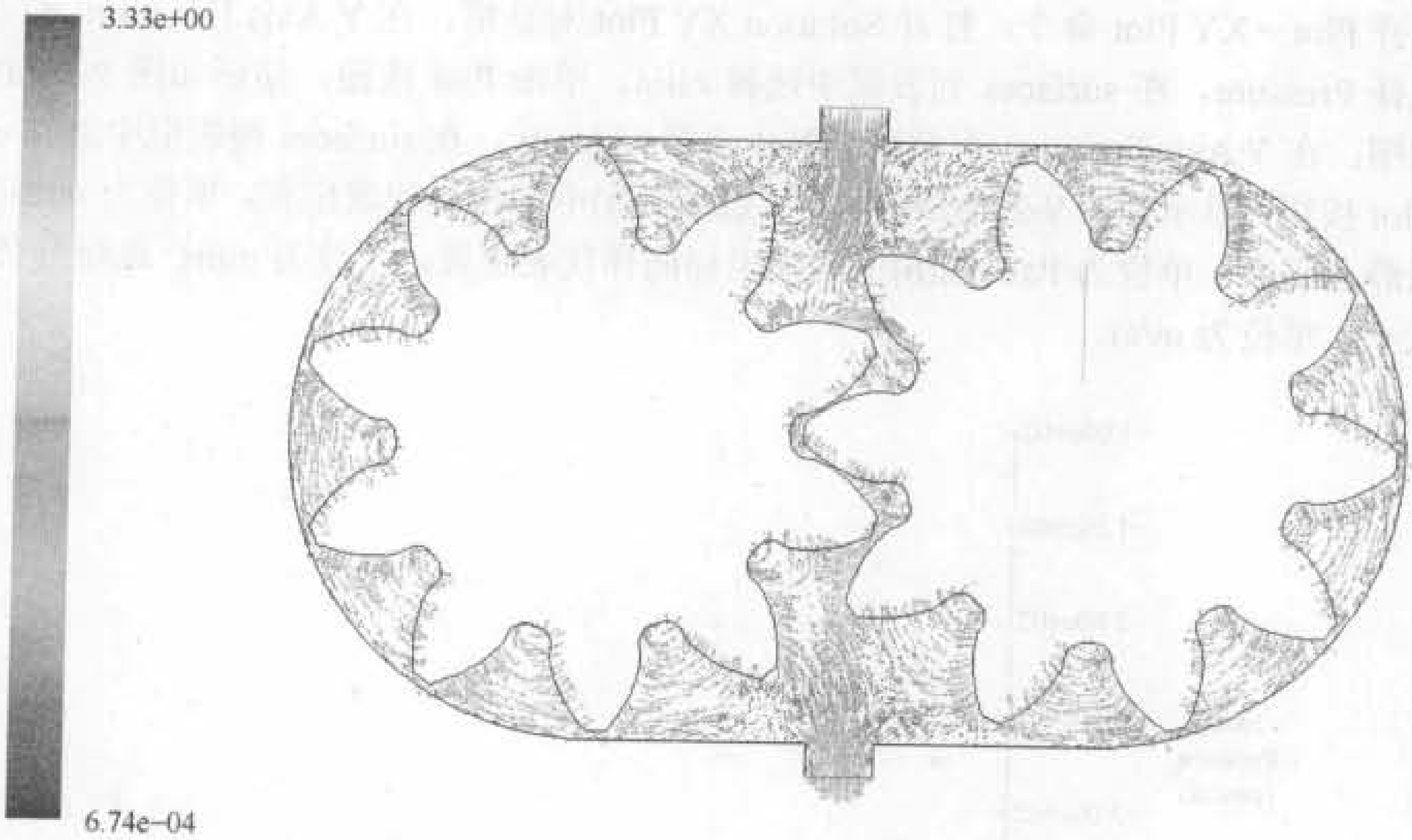


图 9-23　速度矢量图

3)　流线显示

选择 Display→Path Lines 命令，打开 Path Lines 对话框。

在 Step 栏中输入 5000，在 Skip 栏中输入 2，在 Release from Surface 栏中选择 oilin、movewall_left、movewall_right，单击 Display 按钮，显示如图 9-24 所示的流线图。

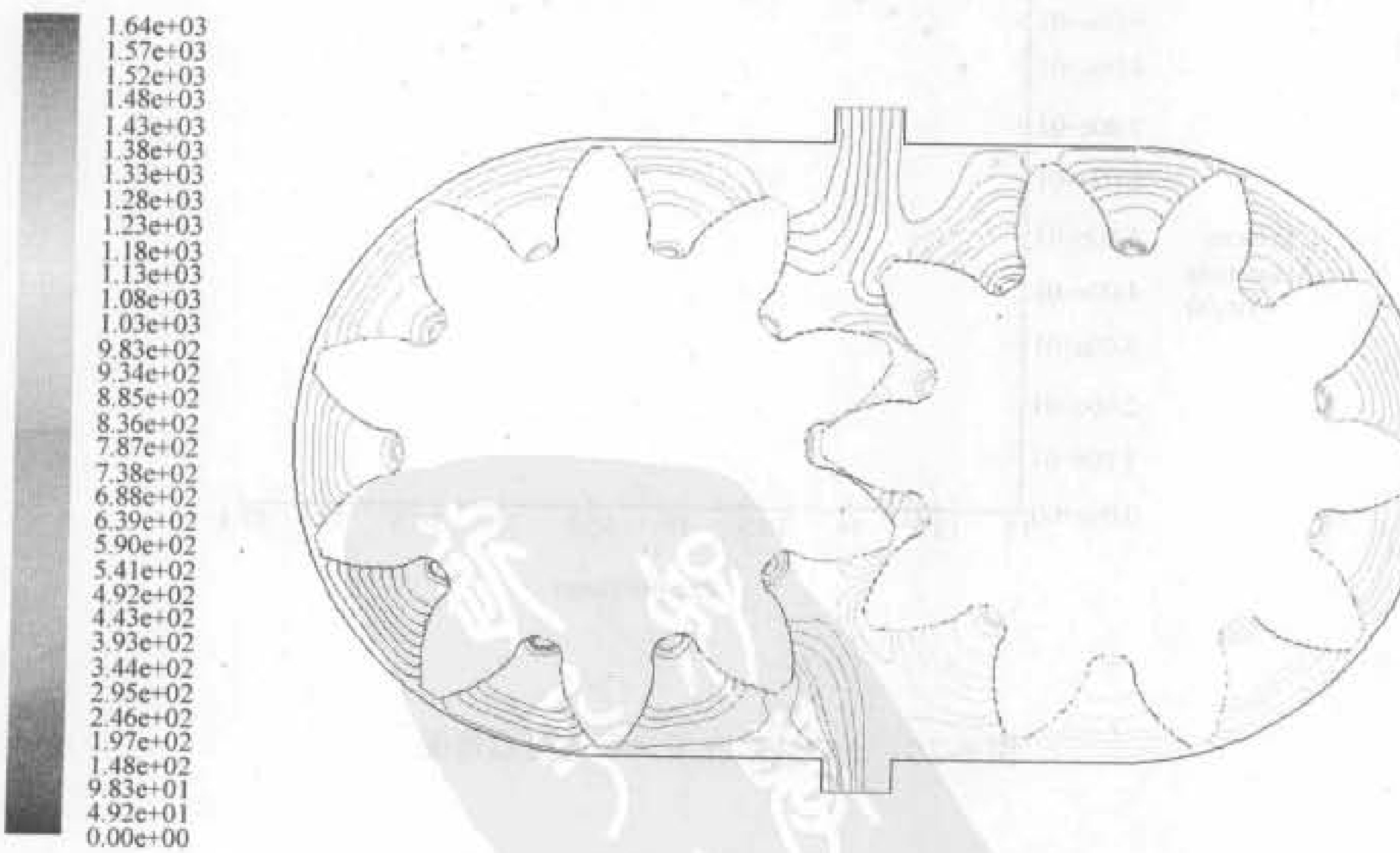
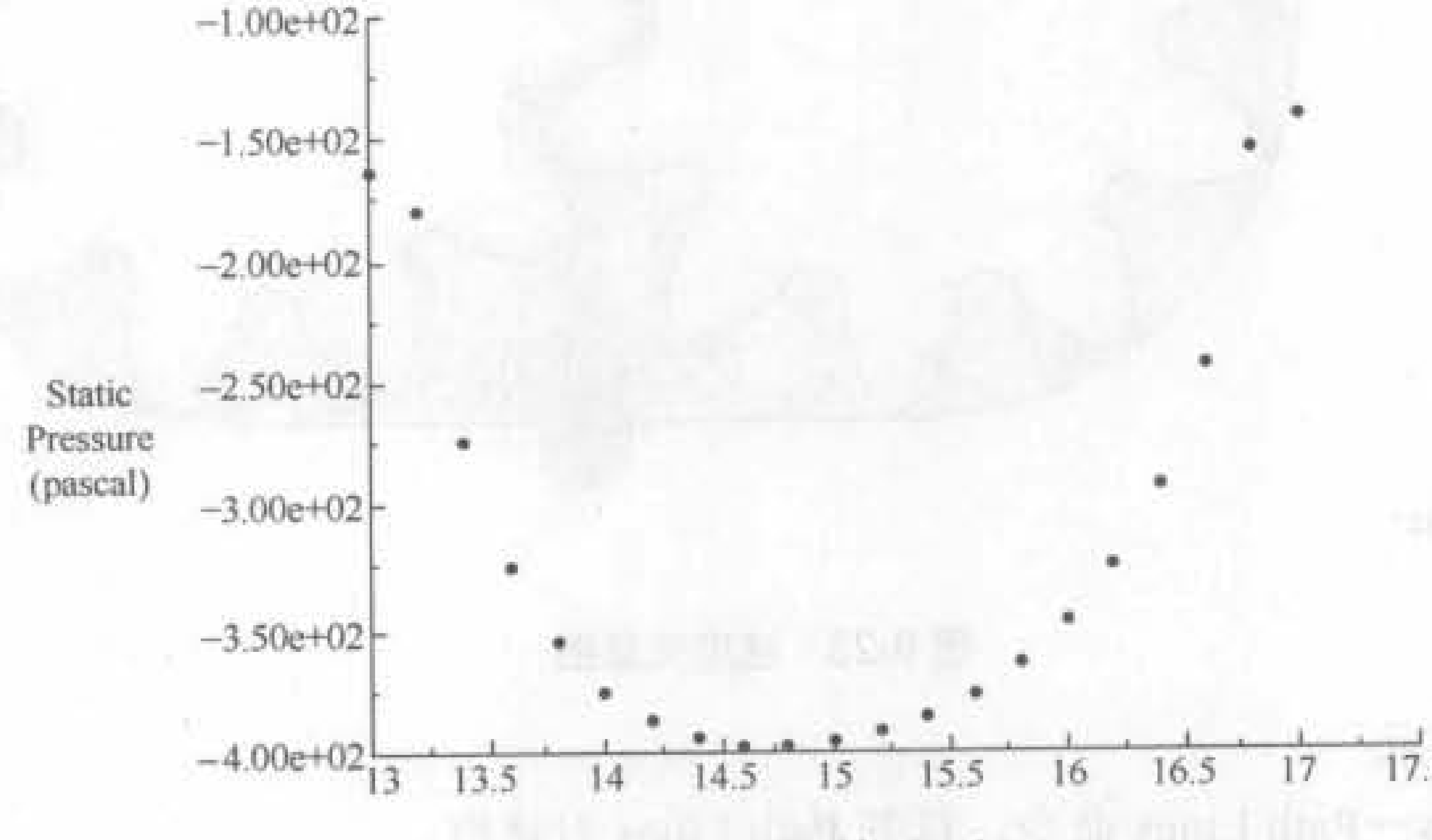


图 9-24　流线显示

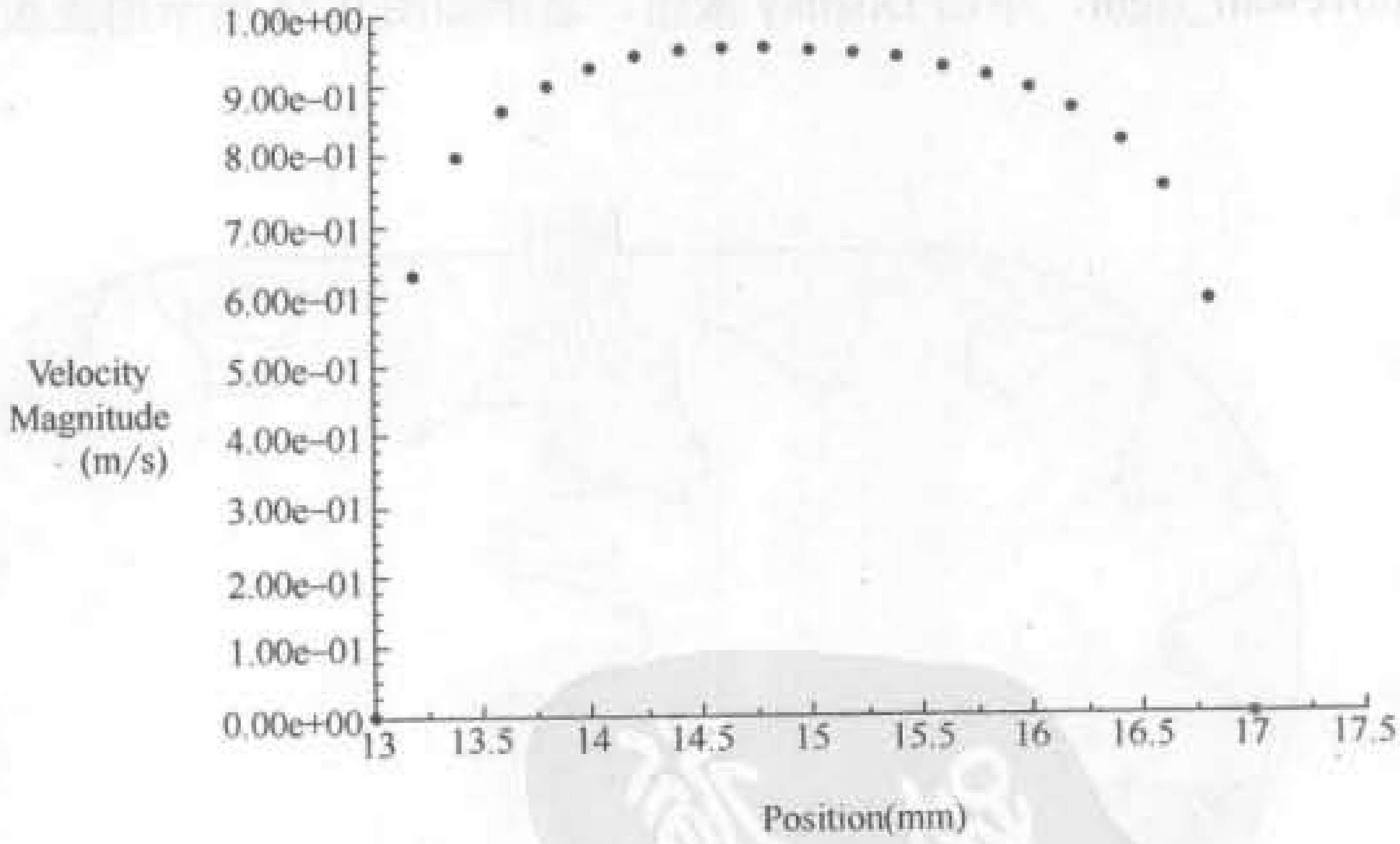
4)　XY 曲线显示

显示计算区域任意线段或任意面上的物理量的变化情况。

选择 Plot→XY Plot 命令，打开 Solution XY Plot 对话框，在 Y Axis Function 下拉列表框中选择 Pressure，在 surfaces 列表框中选择 oilin，单击 Plot 按钮，显示如图 9-25(a)所示的曲线图；在 Y Axis Function 下拉列表框中选择 Velocity，在 surfaces 列表框中选择 oilin，单击 Plot 按钮，显示如图 9-25(b)所示的曲线图(图(a)中横坐标代表位置，单位为 mm；纵坐标代表静态压力，单位为 Pa。图(b)中的横坐标同样代表位置，单位为 mm；纵坐标代表速度的大小，单位为 m/s)。



(a)



(b)

图 9-25 进口线段上压力和速度曲线图

第 10 章　滑移网格基础

在 Fluent 中，除了动网格能够描述计算区域瞬态运动外，还有滑移网格能描述计算区域的运动，是在 MRF 和混合平面法基础上发展起来的。本章主要介绍滑移网格的基本概念及在 Fluent 中进行相关计算的设置。

10.1　滑移网格概述

10.1.1　滑移网格的应用及运动方式

1. 滑移网格的用途

(1) 图 10-1 所示的是为通过同一隧道的两辆汽车或火车。

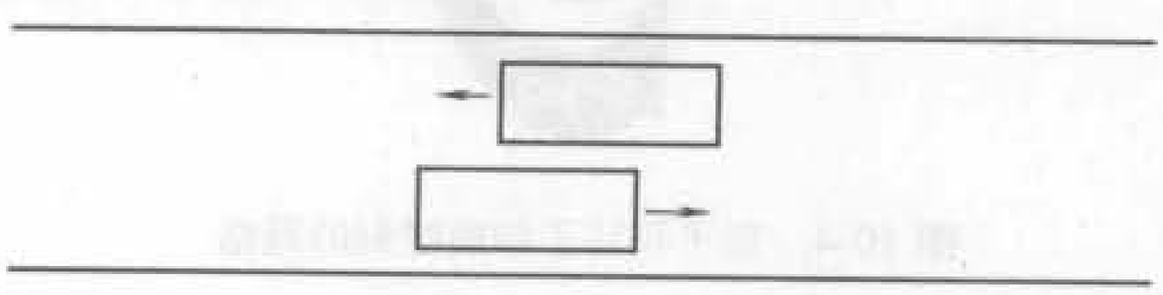


图 10-1　通过隧道的车辆

(2) 图 10-2 所示为定转子的相互作用，图 10-3 所示为风机的运动。

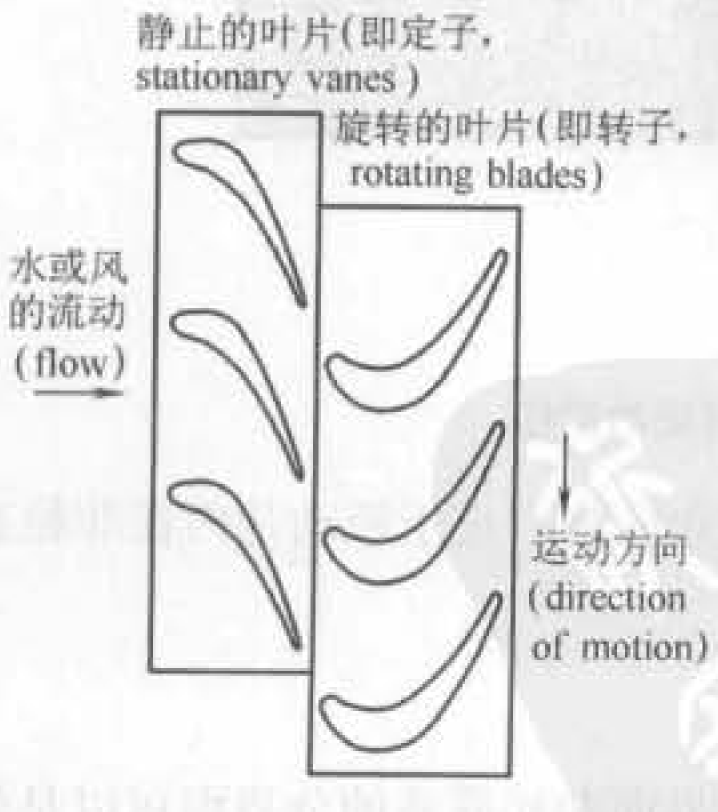


图 10-2　动定子相互作用

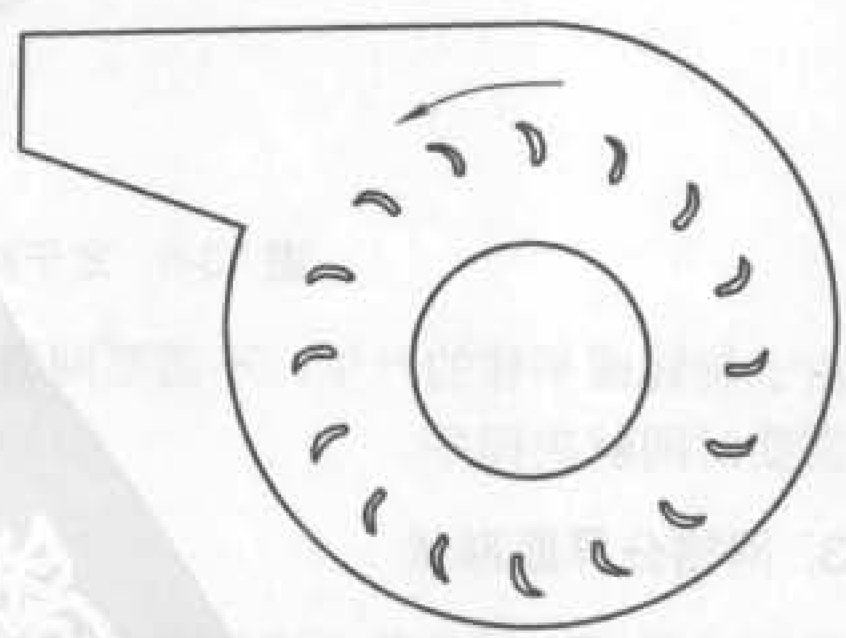


图 10-3　风机

2. 滑动网格的运动

在两个或更多的单元区域(如果在每个区域独立划分网格，则必须在开始计算前合并网格)应用滑动网格技术，每个单元区域至少有一个边界的分界面，该分界面区域和另一单元

区域相邻，相邻的单元区域的分界面互相联系形成“网格分界面”，这两个单元区域互相之间相对沿网格分界面移动。

注意：网格分界面必须定位，因为两边均有流体单元。例如，如图 10-2 所示的几何体的网格分界面必须位于转子和定子之间的流体区域，而不能在转子或定子边缘的任何部分。

在计算过程中，单元区域沿着网格分界面相互之间滑动(如旋转或平移)，而两个区域的网格不会发生变化。图 10-4 和图 10-5 描述了两网格不同时刻的位置，前者是初始时两个网格区域的位置，后者为平移后两网格区域的位置。

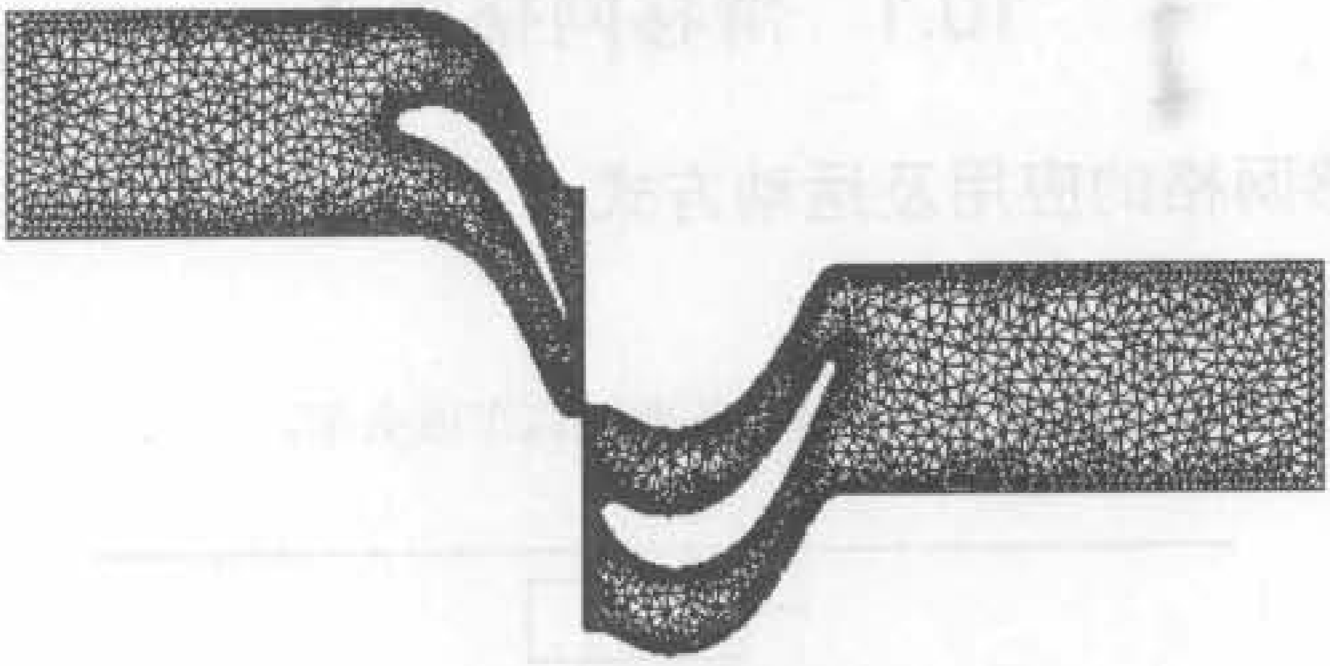


图 10-4　定子和转子初始时刻的网格

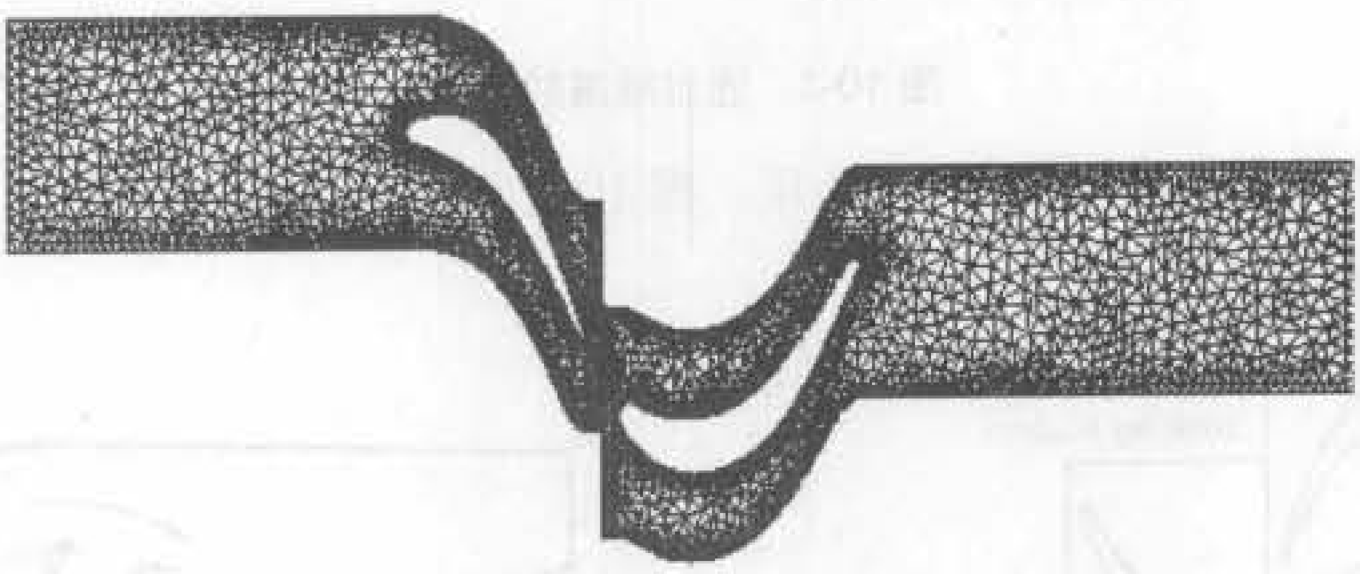


图 10-5　定子和转子相对平移后的网格

由于旋转或平移的产生，不需要网格分界面的节点队列。由于流动是内在非稳态的，所以需要时间解决程序。

3. 网格分界面形状

当两分界面边界是基于同样的几何体时，网格分界面和相联系的分界面可以是任何形状。图 10-6 是线性网格分界面的例子，图 10-7 是圆弧网格分界面的例子，两图的网格分界面以虚线表示。

如把图 10-6 伸展为三维时网格分界面将是平面矩形，如把图 10-7 伸展为三维时网格分界面将是圆柱体。

图 10-8 是圆锥形分界面的应用例子，斜线表示在三维平面上的圆锥形的分界面的交线。

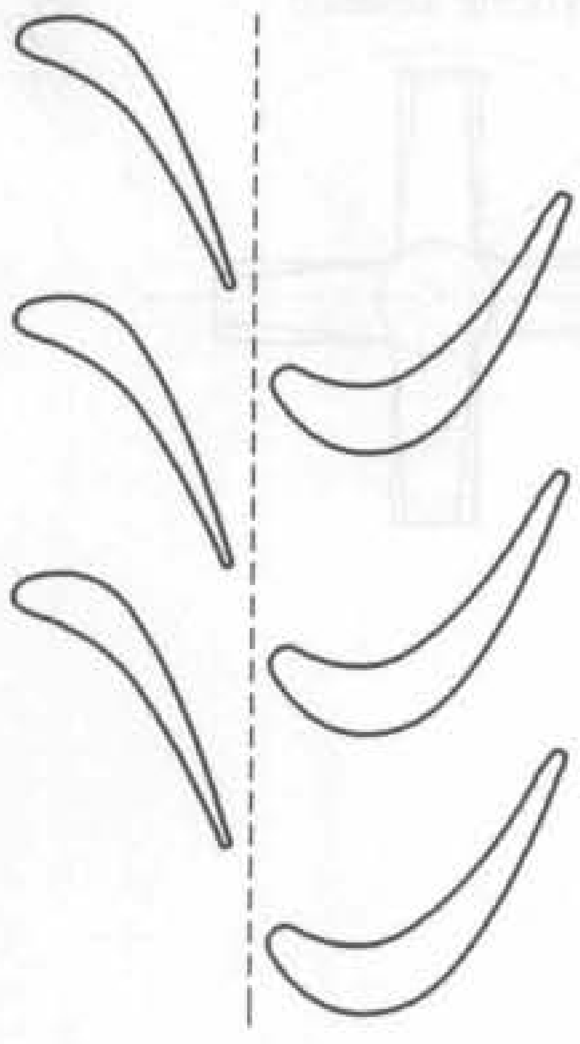


图 10-6　线性网格分界面

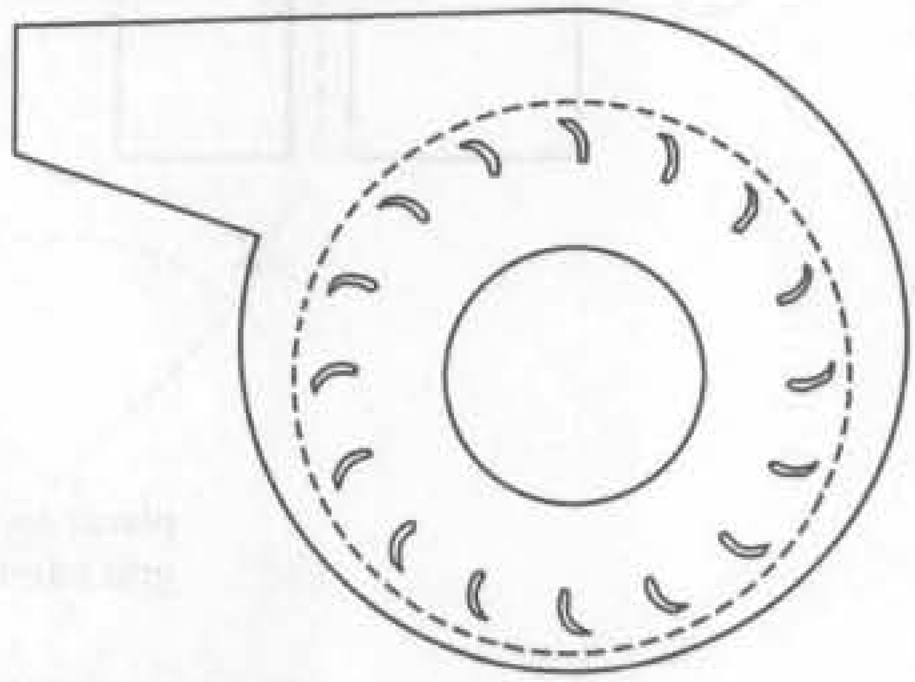


图 10-7　圆弧网格分界面

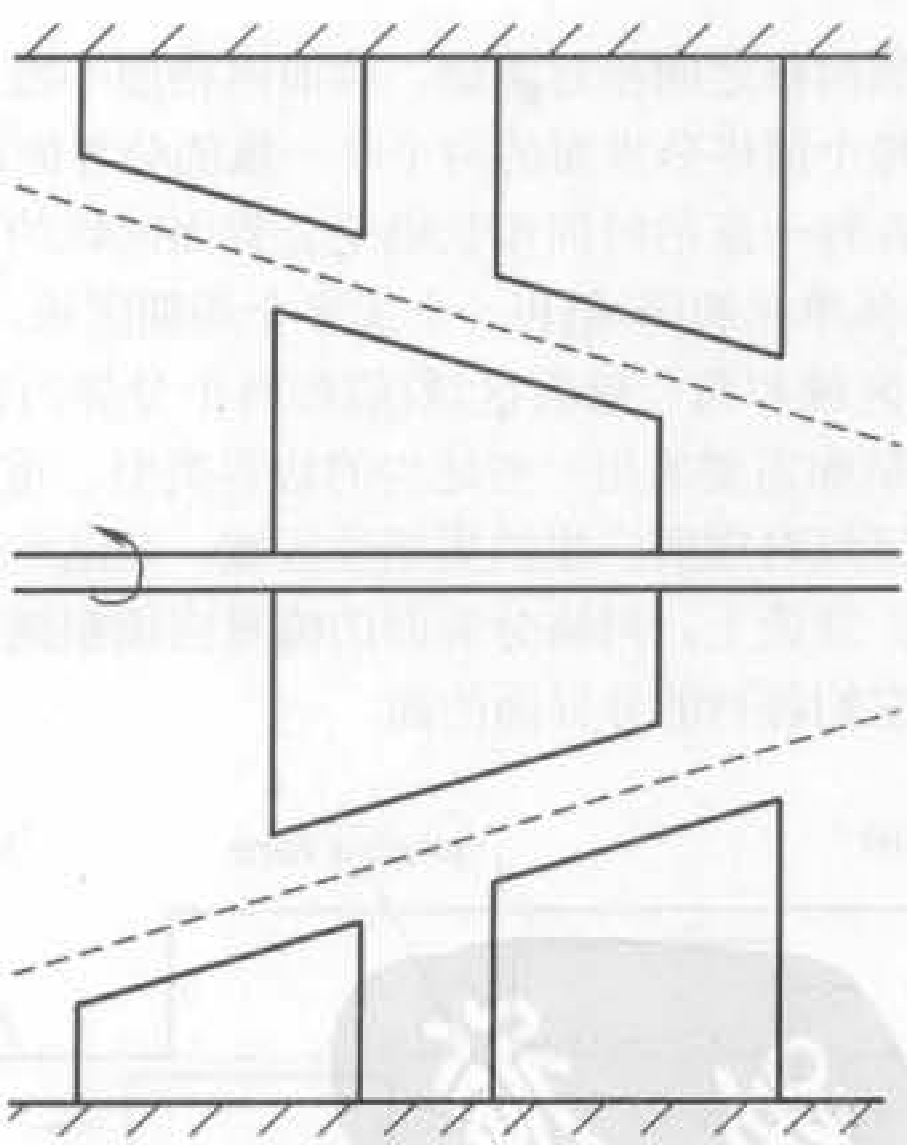


图 10-8　圆锥形网格分界面

对于轴向的转子/定子构造，其旋转和静止部分沿轴线成一行，而不是同中心的(见图 10-9)，分界面是平面扇形。该平面扇形是垂直于旋转轴的截断面的区域，旋转轴沿着转子和定子之间的轴线位置。

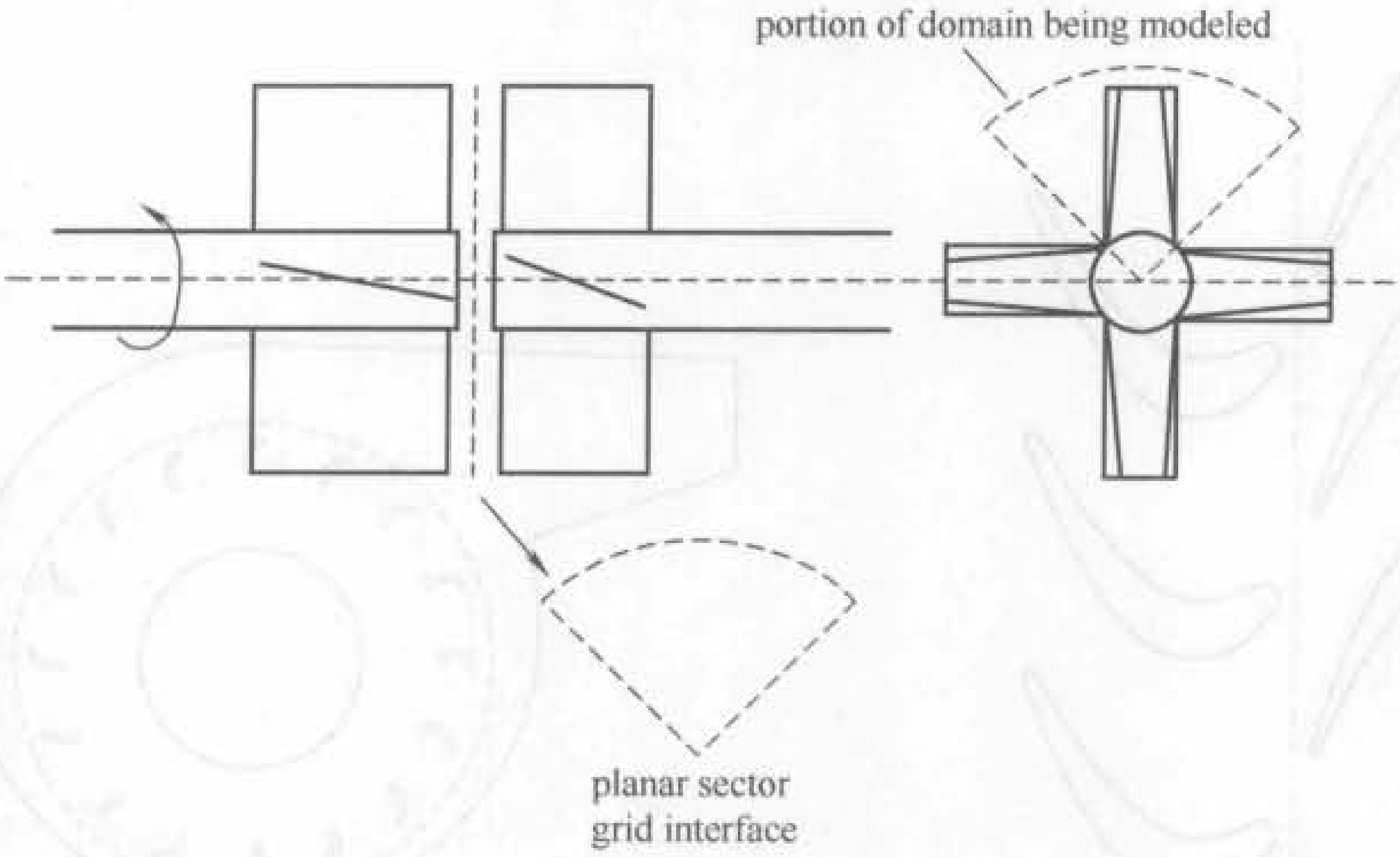


图 10-9 扇形网格分界面

10.1.2 滑移网格的原理

滑移网格模型允许相邻网格之间相对滑动，因而网格面不需要在分界面上排列，对于滑移网格关键是计算流进每个网格分界面的两个非一致的分界面区域。

为了计算界面流动，在每一新的时间步长确定分界面区域的交界处，在该交界面产生了内部区域(在两边都有流体单元的区域)和一个或多个周期区域。如果不是周期性的问题，那么交界面产生一个内部区域和两个壁面区域(假如两个分界面区域完全交界则没有壁面区域)，如图 10-10 所示，壁面需要采用一些适当的边界类型。重叠的分界面区域对应所产生的内部区域；不重叠的区域对应所产生的周期性区域。在这些交界面区域的面的数目随着分界面相对移动而不同。理论上，网格分界面的流量应该根据两分界面的交叉处所产生的面来计算，而不是根据它们各自的分界面的面。

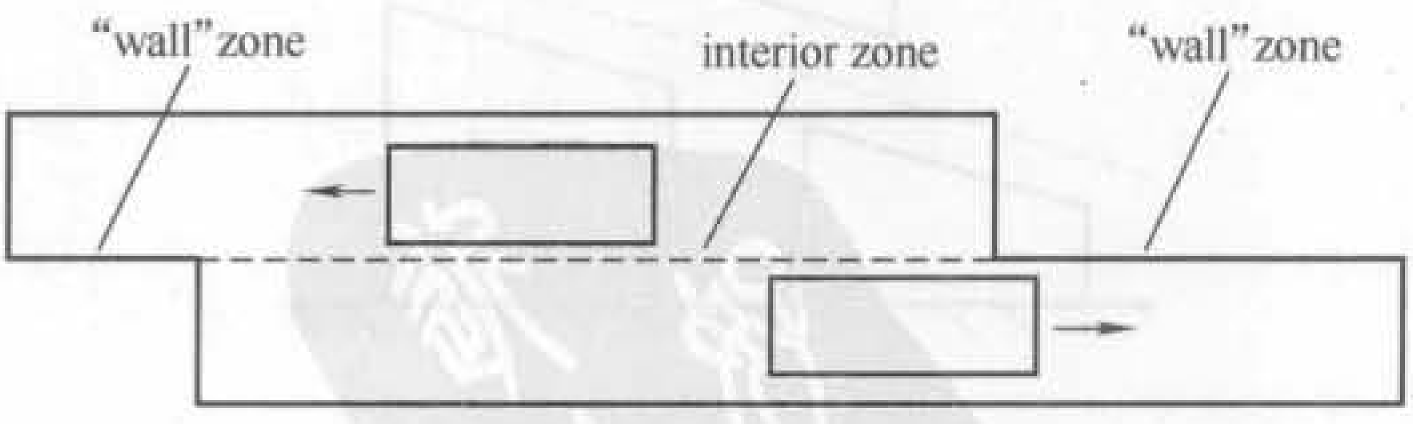


图 10-10 非周期性分界面相交区域

在图 10-11 所示的例子中，分界面区域由 A-B 面和 B-C 面以及 D-E 面和 E-F 面组成，交叉处产生 a-d 面、d-b 面和 b-e 面等。在两个单元区域重叠处产生 d-b 面、b-e 面和 e-c 面而组成内部区域，剩余的 a-d 和 c-f 面成对形成周期性区域。例如，计算分界面流入Ⅳ单元的流量时，用 d-b 面和 b-e 面代替 D-E 面，并从Ⅰ单元和Ⅲ单元各自传递信息到Ⅳ单元。

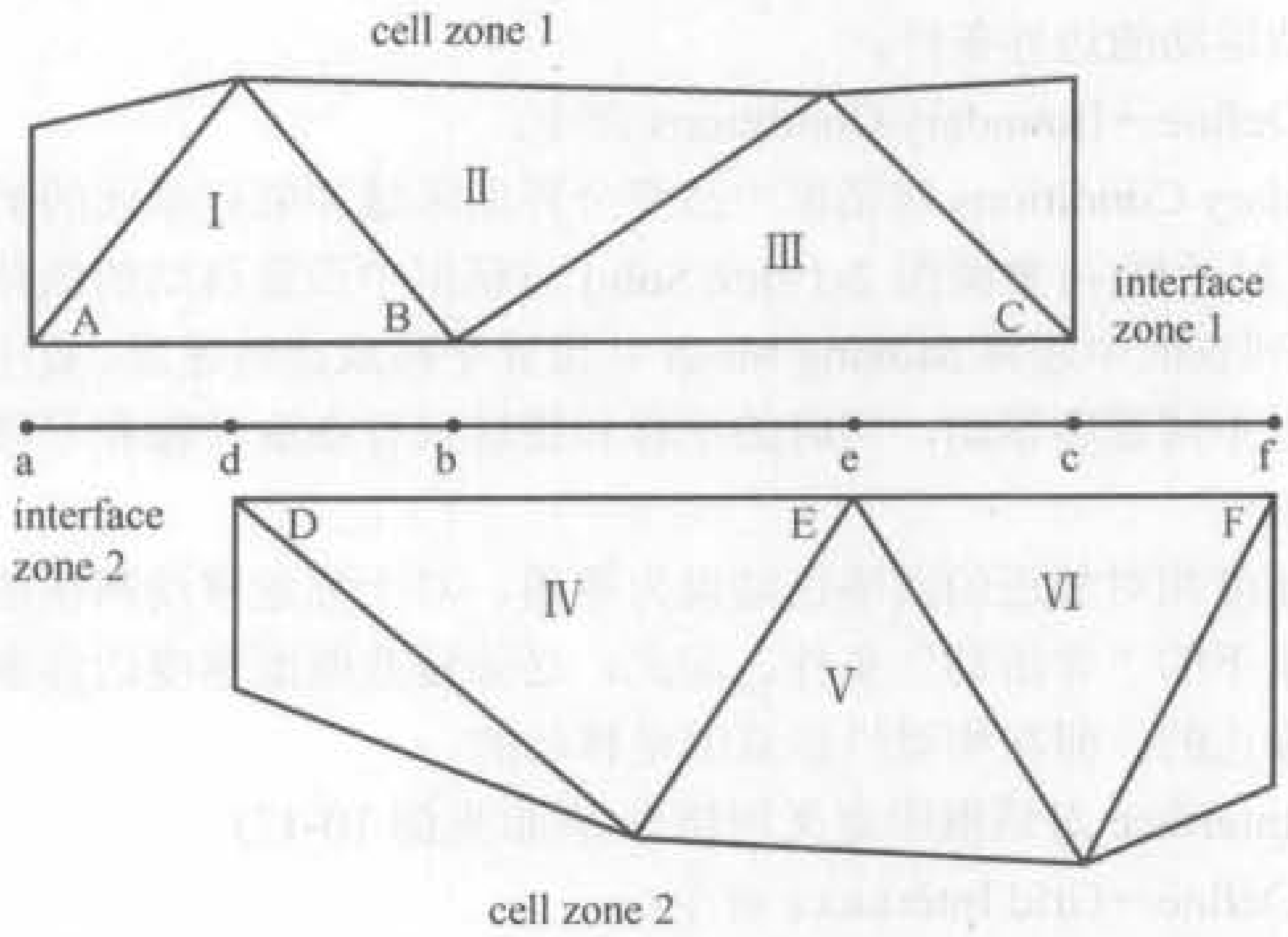


图 10-11　二维网格分界面

10.2　滑移网格的基本设置

10.2.1　网格的前提条件

在 Fluent 问题建立开始之前，确定创建的网格满足以下要求。

(1)　不同的单元区域存在于不同滑动速度的范围的每一部分。

(2)　网格分界面必须位于没有该面的法线运动的位置。

(3)　网格分界面可以是任意形状(包括三维曲面)，一般假定两分界面的边界是基于同一几何体。同时，如果网格有明显特征(如 90° 角)，则分界面两边几乎遵循该特征。

(4)　如果创建的是多单元区域的单一网格，必须保证每个单元在滑动边界有明显的面域。相邻单元的面有一样的位置和形状，但每个面对应于各自的一个单元；也可以每个单元各创建一个独立网格文件，然后再在 Fluent 中合并它们。

(5)　如果是用周期性模拟一个转子/定子几何体，转子叶片网格的周期角度必须和静止叶片的角度相等。

(6)　创建网格分界面之前必须正确定位所有的周期性区域(旋转的或者平移的)。

(7)　对于三维情况，如果分界面是周期性的，那么只有一对周期边界和分界面相邻。

根据上述要求，在具体建模时，对于交界处，应该设置两个面，而后在 Gambit 中分别指定边界类型(可以先设定为 wall 类型)，以便在 Fluent 中进行交界面的设置。

10.2.2　问题的建立

建立滑动网格问题的步骤如下(注意：该过程只包括滑动网格模型的必要步骤，其他模型、边界条件等也需如常建立)。

(1)　在 Solver 对话框中激活非稳态流动模型的适当选项。

操作：选择 Define→Models→Solver 命令。

(2) 设置滑动运动的边界条件。

操作：选择 Define→Boundary Conditions 命令。

① 在 Boundary Conditions 对话框中改变分界面区域和邻近单元的类型为 interface。

② 在 Fluid 对话框(可参阅图 2-19)或 Solid 对话框中设置移动的流体或固体区域，在 Motion Type 下拉列表框中选择 Moving Mesh 并设置平移或旋转速度。要注意固体区域和邻近流体区域不能以不同速度移动，同时的平移和旋转只有在旋转轴和平移方向一致时才能模拟(如原点固定)。

默认的壁面速度相对邻近的网格运动设为零值，对于邻近移动网格的壁面来说结果是一个在网格参考系下的“非滑动”条件。因此，必须修改壁面速度边界条件，除非在绝对坐标系中壁面是静止的，即在相对坐标系中是移动的。

(3) 在 Grid Interface 对话框中定义网格分界面(见图 10-12)。

操作：选择 Define→Grid Interfaces 命令。

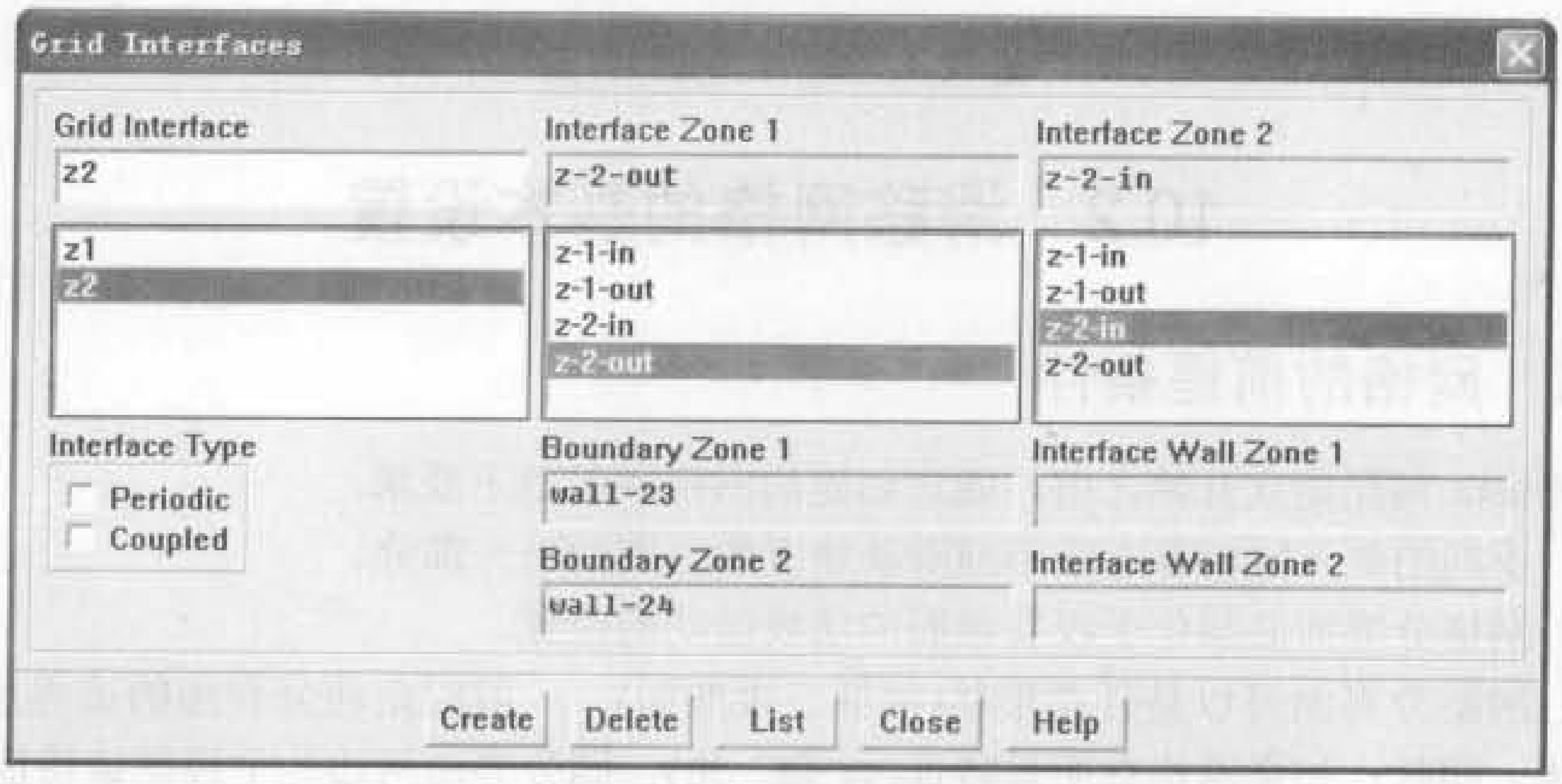


图 10-12 网格分界面设置对话框

① 在 Grid Interface 栏中输入分界面的名字。

② 确定组成网格分界面的两分界面区域，在 Interface Zone 1 和 Interface Zone 2 列表中各选择一个，次序无关。

③ 设置 Interface Type，如果适当，有两个选项：对于周期性问题，选中 Periodic 复选框；分界面位于固体和流体区域，选中 Coupled 复选框。

④ 单击 Create 按钮，建立新的网格分界面。

对于所有分界面类型，Fluent 创建了分界面的边界区域，如 Wall-23 和 Wall-24，在 Interface Zone 1 和 Interface Zone 2 下出现。可使用 Boundary Conditions 对话框改变它们的区域类型，例如 pressure far-field、symmetry 和 pressure outlet。

如果创建了一个错误的网格分界面，可以在 Grid Interface 列表框中选择该分界面，单击 Delete 按钮删除它，删除分界面同时也删除了任何边界区域。